DATA-DRIVEN TRANSFORM OPTIMIZATION FOR NEXT GENERATION MULTIMEDIA APPLICATIONS

A Thesis
Presented to
The Academic Faculty

by

Osman Gokhan Sezer

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
School of Electrical and Computer Engineering

Georgia Institute of Technology
December 2011

DATA-DRIVEN TRANSFORM OPTIMIZATION FOR NEXT GENERATION MULTIMEDIA APPLICATIONS

Approved by:

Professor Yucel Altunbasak, Advisor
School of Electrical and Computer Engineering
Georgia Institute of Technology

Research Asst. Prof. Onur Guleryuz
Department of Electrical Engineering
Polytechnic Institute of NYU

Professor Jim McClellan
School of Electrical and Computer Engineering
Georgia Institute of Technology

Professor Xiaoming Huo
School of Industrial and Systems Engineering
Georgia Institute of Technology

Professor Justin Romberg
School of Electrical and Computer Engineering
Georgia Institute of Technology

Professor David Anderson
School of Electrical and Computer Engineering
Georgia Institute of Technology

Date Approved: 15 Aug 2011

To my parents,

Annem ve rahmetli Babama..

ACKNOWLEDGEMENTS

I would like to thank my advisor Dr. Yucel Altunbasak for his support and encouragements on personal and professional levels. I feel that I am lucky to become a member of his team, which brought me invaluable friendships and experience. I am grateful to have the opportunity to collaborate with Dr. Onur Guleryuz both during my internship and afterwards. I would like to express my sincere gratitude to Dr. Justin Romberg, even brief discussions with him helped me to steer my research to more a fruitful path. It has been great pleasure to be a part of Center for Signal and Image Processing (CSIP) and TI Leadership University Program with the leadership of Dr. Jim McClellan, like many of us I am thankful to him.

I would like to thank Dr. David Anderson and Dr. Xiaoming Huo to honor us by serving in my thesis committee. I wish to express my deepest gratitude to my friends at CSIP for their support and joyful conversations. I will cherish those memories in Georgia Tech for the rest of my life.

Finally, I am grateful to my late father, Ismail Sezer and my mother, Ayfer Sezer for unwavering support throughout my life, which kept me going. I am also thankful to my sister Aylin Celen for everything she has done to take care of our family in my absence.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

SUMMARY

The objective of this thesis is to formulate a generic dictionary learning method with the guiding principle that states: Efficient representations lead to efficient estimations. The fundamental idea behind using transforms or dictionaries for signal representation is to exploit the regularity within data samples such that the redundancy of the representation is minimized subject to a level of fidelity. This observation translates to rate-distortion cost in compression literature, where a transform that has the lowest rate-distortion cost provides a more efficient representation than the others.

In our work, rather than using as an analysis tool, the rate-distortion cost is utilized to improve the efficiency of transforms. For this, an iterative optimization method is proposed, which seeks an orthonormal transform that reduces the expected value of rate-distortion cost of an ensemble of data. Due to the generic nature of the new optimization method, one can design a set of orthonormal transforms either in the original signal domain or on the top of a transform-domain representation. To test this claim, several image codecs are designed, which use block-, lapped- and wavelet-transform structures. Significant increases in compression performances are observed compared to original methods. An extension of the proposed optimization method for video coding gave us state-of-the-art compression results with separable transforms. Also using the robust statistics, an explanation to the superiority of new design over other learning-based methods such as Karhunen-Loeve transform is provided.

Finally, the new optimization method and the minimization of the "oracle" risk of diagonal estimators in signal estimation is shown to be equal. With the design of new diagonal estimators and the risk-minimization-based adaptation, a new image

denoising algorithm is proposed. While these diagonal estimators denoise local image patches, by formulation the optimal fusion of overlapping local denoised estimates, the new denoising algorithm is scaled to operate on large images. In our experiments, the state-of-the-art results for transform-domain denoising are achieved.

CHAPTER I

INTRODUCTION

This thesis presents the design of rate-distortion-optimized transforms for next-generation multimedia applications. The fundamental idea behind using transform coding is to exploit the regularity within data samples such that the redundancy of the representation is minimized subject to a fidelity cost. However, due to the non-stationarity of image, speech and audio signals the local statistics (hence the regularity) vary significantly across the data, which urges transform adaptation for efficient representation.

Data representation with transforms has been used in various signal processing, reconstruction and compression applications. One can split the literature into two categories. The first category of transform design methods assumes certain regularity within the data samples, and builds a model on the sample variations in a local neighborhood. By using the harmonics or certain smoothness characteristics, the model-based methods are able to condense signal characteristics into a few transform coefficients. The Fourier transforms, the wavelets and the lapped transforms, even DCT can be included in this category [4, 3, 50]).

Depending on the characteristics of the signal, the model-based approximations to signals can achieve optimal representation. The wavelet transform, which provides the optimal representation for piecewise-smooth $1-$D signals, is a good example. The efficiency of a transform representation over a class of data/signals may change based on the characteristics of the signal. A transform that enables a high fidelity reconstruction with only a few non-zero coefficients is said to be an efficient representation for that class of signals. In the current literature, the efficiency of a transform representation become synonymous with the sparsity of the representation, and the

transforms that make sparse representation possible are sought after in various signal types in different research fields.

The second category of transform design approaches seeks for the transforms that can represent a particular class of signals with only a few active (non-zero) coefficients. In the model-based methods there is a top-down design regarding the relation of data samples and the corresponding coefficients. However, by imposing sparsity on the coefficients via statistical or deterministic processes [7, 2], the second category is a bottom-up strategy, where the desired coefficient properties are known and imposed on to transforms. In Chapter 2, a detailed review and history of the transform representation and the transform coding is provided for the second category of design algorithms together with a probabilistic framework.

This thesis positions itself close to the second category of design for transform representations, where sparsity is enforced on the coefficients of the representation via $\mathcal{L}_0$ norm minimization, deterministically. Since our focus will be on orthonormal transforms, without the need of $\mathcal{L}_1$ minimization, we are able to formulate an algebraic method that can improve the representation efficiency of transform over a class of signals. In Chapter 3, a generic transform learning method, which can be applied to signals with different statistical characteristics, is proposed. To demonstrate the efficiency of the new representation, several image compression experiments are designed. These experiments are targeted to reveal how the same algebraic optimization can be reused to improve the representation efficiency of transforms for the signals with different characteristics. In one image codec, wavelet coefficients are treated as the original signal and a set of orthonormal transforms over wavelet coefficients are learned such that the rate-distortion performance of the wavelet transform is improved. A similar idea is applied to lapped orthogonal transforms, and a consistent improvement of compression performance is observed in all test images at all bitrates.

In the Chapter 4, rather than applying a set of orthonormal transforms on the

top of wavelets to have sparse representation, a new adaptive lifting-based image compression method is described. Lifting is an algorithm that is based on prediction of data samples from each other, and any wavelet decomposition is shown to be factored into a couple of lifting steps [20]. Chapter 4 focuses on a special case of bi-orthogonal wavelet transforms (BWT) called Boxcar/Wavelet decomposition, which uses dyadic averages and their interpolations. The new adaptive lifting algorithm makes use of this structure and replaces the 1-D interpolators with a set of trained 2-D filters. The idea is to have a nonlinear prediction (interpolation) that will adapt to the context around the low-pass wavelet coefficients such that the energy in the high-pass bands is reduced. In general, 20 context classes are defined and the corresponding filters are found by minimizing a least-square cost. These filters are observed to have directional characteristics with some textural clues. Although the proposed architecture was 1/3-tap, experiments show competitive subjective and objective results with popular 9/7-tap and 5/3-tap BWTs.

Another target of this thesis is to design new transforms to improve the efficiency of video coding. The Discrete Cosine Transform (DCT) plays a vital role in the development of video compression standards, due to ease of its application and the coding efficiency it provides. For next-generation video coding, a new set of 2-D separable transforms has emerged as a candidate to replace the DCT. These separable transforms are learned from residuals of each intra prediction mode; hence termed as Mode dependent- directional transforms (MDDT). MDDT uses the Karhunen-Loeve Transform (KLT) to create sets of separable transforms from training data. Since the residuals after intra prediction have some structural similarities, transforms utilizing these correlations improves coding efficiency. However, the KLT is the optimal approach only if the data has a Gaussian distribution without outliers. Due to the nature of the least-square norm, outliers can arbitrarily affect the directions of the KLT components. In Chapter 5, we will address robust learning of separable transforms by

enforcing sparsity on the coefficients of the representations. With this new approach, it is possible to improve upon the video coding performance of H.264/AVC by up to 10.2% BD-rate for intra coding. At no additional cost, the proposed techniques can also provide up to 3.9% improvement in BD-rate for intra coding compared to existing MDDT schemes.

Increased efficiency provided by the new representation has a reflection in the signal estimation problems as well, which is studied in Chapter 6. First, the signal estimation problem and its basics are given such as the description of an estimator, the risk of an estimator, the bounds on estimator risks, and approximations to risks. Basically, the estimators are linear or non-linear operators that estimate the original signal given a noisy observation of the signal. In Chapter 6, we have used diagonal estimators, which are shown to be close to the optimal estimators provided that the signal is well approximated in the given orthogonal domain [27]. Later, we show that learning transforms to minimize the risk of estimation is equivalent to the original formulation in previous chapters, which is designed to increase the representation efficiency of the transforms.

To test the performance, a new image denoising algorithm is proposed in Chapter 6, which adaptively chooses an orthonormal transform learned via the new method. Two important contributions of our denoising algorithm apart from the dictionary learning method are the adaptation of the transforms and the fusion of the local denoised estimates to generate a global and final signal reconstruction. Assuming that we have learned a set of orthonormal transforms that are tuned to minimize the estimation risk of a class of signal (such as blocks with certain gradient feature), the adaptation is performed by seeking a transform in that set with the minimum risk score for that particular signal. The idea is that the transform (and the corresponding estimator), which is optimized for a certain class of signals, is expected to give the lowest risk score compared to the other transforms in the set for a block of the same

class. Thus, the estimation risk is used for the transform adaptation.

The second contribution of the new denoising method is explaining the optimal fusion of local estimates of a signal to obtain the final reconstruction. The need for fusion of estimates comes from the denoising approach that is utilized. Basically, the denoiser estimates the neighborhood of each pixel, hence due to the overlap, there will be multiple estimates for the same pixel/sample location. Guleryuz has noted that some of these estimates are better than the others and deserve higher weight [35]. Here, we have arrived the same conclusion, and found that the risk of the estimators can be used as an approximation to the optimal weights. Our formulations and the algorithm confirm the importance of the estimation risk in transform learning, transform adaptation, and fusion of the estimates for the denoising problem.

1.1 Summary of the Contributions

It is useful to clearly list the contributions of the presented thesis that we think are important.

⋆ Contributions of Chapter 3- Sparse Orthonormal Transforms

- A new generic data-driven transform optimization method is described in details with the theory and the implementations.
- A block-based image codec is designed, which makes use of the new transforms for image coding. Consistent increase in bitrates compared to Discrete Cosine Transform (DCT) based image codec is observed with up to 1dB improvement.
- In another block-based codec, the transform sizes are adaptively changed with a quadtree segmentation. Up to 2dB improvement is observed in natural images and up to 6dB improvement is observed for for synthetic images.
- A new lapped transform is create with the new learning algorithm. On the top of the standard lapped bi-/orthogonal transform, a new set of directional

transforms are learned. A consistent coding efficiency is gained over the standard lapped transform with up to 0.8dB improvement. This implementation is also one of the first directional lapped transform designed in the literature, which does not requires knowledge of complex modulation algorithms.

• Similar to wedge- and foot-prints methods, the coefficients of wavelets are mapped to a sparser domain. Rather than using fixed models, a set of orthonormal transforms are designed and applied on the top of wavelet decomposition. In the smooth image regions, where the wavelet decomposition work, are kept unchanged. Around the directional edges new orthonormal transforms provided a sparser representation by utilizing correlation between the wavelet coefficients. Again, a consistent increase in rate-distortion performance is observed compared to the original wavelet decomposition.

⋆ Contributions of Chapter 4- Training-based 2-D Nonlinear Lifting

• A new nonlinear wavelet decomposition algorithm is presented that replaces the prediction step of the lifting algorithm with a more complex 2-D interpolator that is designed to adapt the local context of the image.

• The local context is determined by extracting features from low-pass coefficients of the proposed decomposition algorithm. Similar to the interpolation with resolution synthesis method [5], a 2-D filter is learned for each context class. Subjective gains are observed around edges.

⋆ Contributions of Chapter 5- Mode Dependent Sparse Transforms for Video Coding

• A novel separable filter design technique based on Chapter 3 is introduced for video coding. In the new design for each encoding mode a vertical and horizontal filter is learned by enforcing sparsity on the coefficients

• The difference between the proposed transform design algorithm and Karhunen-Loeve transform (KLT) is explained based on robust statistics. This is done by examining the error norms of the KLT and the proposed method. We have revealed that due $\mathcal{L}_0$-norm regularization, the cost function (or the error norm, or M-estimator) of the proposed method reduces the influence of the outliers in the data. Robustness claims are supported by simple experiments provided in this chapter.

• When incorporated into a video codec, the new 2-D separable transforms are observed to produce state-of-the-art results.

⋆ Contributions of Chapter 6- Risk-Minimizing Transforms for Signal Estimation

• Using the oracle risk, a risk minimization framework is described. The oracle risk of a diagonal estimator is used to find the upper and lower limits of the actual estimation risk. Once the oracle risk of an estimator is minimized over a class of signal, it is expected to improve the estimation performance. This is achieved by reformulation the original transform optimization given in Chapter 3 into a risk minimization problem, where we seek for transforms (any corresponding estimators) that will reduce the oracle risk.

• With this framework, a set of transforms (or estimators) are learned and adaptively applied over a noisy data. The adaptation is done based on the risk of the estimators (the estimator that gives minimum risk value for that particular block of signal is selected).

• Together with this new adaptation, first the optimal fusion of local estimates is formulated and then a risk-based approximation is proposed to implement the new data fusion technique. Since we are using block transforms, the denoising operation (or estimation process) is performed per block. From local

estimates, a global signal reconstruction is needed. Generally, averaging of the estimates are done to reconstruct the final denoised signal from the denoised blocks. Here, we have presented the theory and the implementation of optimal weighted averaging to improve the overall signal estimation efficiency.

• We have also formulated how to fuse estimators with different support sizes for reconstructing the final denoised signal. The image denoising algorithm that is based on the new estimators and the adaptive support size selection shows significant estimation gains compared to dictionary-based denoising methods.

CHAPTER II

ORIGIN AND HISTORY OF THE PROBLEM

Data representation with transforms has been used in various signal processing, reconstruction, and compression applications. Starting with Fourier, the search for the best transform-domain representations evolved into finding the sparsest one for given data. Transform design follows two fundamental approaches: (1)- model-based transform design (wavelets, lapped transforms, DCT [4, 3, 50]), and (2) data-driven transform design (independent component analysis, K-SVD[7, 2].) While the model-based approaches exploit regularity within data samples by using mathematical models of smoothness, the data-driven methods focus on the statistics of an ensemble of signals and generate dictionaries that adapt to the characteristics of the signals.

In this thesis, although we offer a new approach that integrates model-based and data-driven methods for image coding, our main contribution is in the data-driven dictionary learning process. Therefore, the overview covers a brief history of prior architectures on dictionary learning. The search for dictionaries to explain environment, particularly for image data, can be traced back to early human vision research, which evolved into sparse coding ideas.

2.1 A Perspective from Human Vision Research

First attempts to understand the human vision system and the birth of experimental psychology, later called "psychophysics," dealt with changes in the mental state due to given input to the brain (a black box) such as a light beam. These studies were later supported by neuro-physiological works, and after Hubel and Wiesels pioneering experiments [37, 38], growing attention of scientists focused on the properties of the neurons that act as receptive fields in the primary visual cortex. It was shown in these

experiments that these receptive fields are localized in time and space, have band-pass characteristics both in the spatial and temporal domains, and are selective to certain orientations. In line with Barlow's proposition [6], these receptive fields can be said to act like some redundancy reduction mechanism such that the factorial coding of the input data is achieved. Merging factorial coding and oriented-edge selective receptive field ideas, Field [32] claimed that these receptive fields enable sparse representation of the input data. Thus, only a few features need to be active for representing an image, and for a group of images, a particular feature will rarely be active. This theory later was tested experimentally by Olshausen and Field [53, 52] by using a network that maximizes the sparseness of the input data coming from patches of natural images. Then, Bell and Sejnowski [7] used independent component analysis (ICA), which aims to search for factorial coding of the data by finding linearly independent components. Basically, this is achieved by maximizing the mutual information between an ensemble of signals from the environment and the vectors of a dictionary. It is shown in [52] that ICA and maximization of sparseness for input data actually are related. Afterward, van Hateren and van der Schaaf [73] quantitatively compared the properties of independent component filters and receptive fields in the primary visual cortex. They showed that the properties of the independent component filters obtained by ICA on a large set of natural images resemble properties of the receptive fields of simple cells in the macaque monkeys cortex, which indicates that expected statistics of the natural stimuli in the environment affect the characteristics of receptive fields. Although the independent component model lacks many aspects of simple cells such as contrast adaptation and nonlinearities in orientation tuning, it has clear information-theoretic conclusions based on the statistics of stimuli. Therefore, it can be said that receptive fields work to decompose and reduce the information redundancy in the scene that falls onto the retina for different specialized tasks such as edge detection or contrast adjustment.

The importance of sparse coding and designing dictionaries that will reduce the redundancy in representation lead to the desirability of having overcomplete dictionaries [52]. To solve the limited structural diversity of basis vectors of ICA, Olhausen and Field's proposal to use an overcomplete set of learned basis vectors was important. The signal processing community had already shown that overcomplete representations can provide superior compression and denoising performances [48, 18]. Following these research efforts, a diverse set of methods was proposed for sparse coding and dictionary design, and with the help of recent developments in optimization techniques, successful results have been achieved in various applications [40, 41, 2, 13].

2.2 Probabilistic Framework

To provide a better understanding of learning-based dictionary design, the probabilistic reasoning in [51, 52, 44] lays the foundation. The probabilistic model starts with a generative explanation of observed data:

$$\mathbf{x} = \Phi\alpha + \nu \tag{1}$$

where $\mathbf{x}$ is an observed signal, α is its sparse representation with transform Φ, and ν is the noise term. Let $\mathcal{S}$ be a training set of signals (e.g. image blocks); then, the goal is to find a basis Φ^* that will maximize the likelihood $P(\mathcal{S}|\Phi)$:

$$\Phi^* = \arg\max_{\Phi} P(\mathcal{S}|\Phi). \tag{2}$$

With the assumption that each observation, $\mathbf{x}^j$, is drawn independently, we get

$$P(\mathcal{S}|\Phi) = \prod_{\forall j} P(\mathbf{x}^j|\Phi). \tag{3}$$

Then, the maximum likelihood (ML) expression becomes

$$\Phi^* = \arg\max_{\Phi} \prod_{\forall j} P(\mathbf{x}^j|\Phi). \tag{4}$$

Similarly, one can rewrite the ML function with the help of a logarithmic function as

$$\Phi^* = \arg\max_{\Phi} \sum_{\forall j} \log(P(\mathbf{x}^j|\Phi)). \tag{5}$$

To proceed with the analysis, $P(\mathbf{x}^j|\Phi)$ is expressed in terms of its transform-domain representation (coefficients) α^j as follows:

$$P(\mathbf{x}^j|\Phi) = \int P(\mathbf{x}^j, \alpha^j|\Phi) d\alpha^j = \int P(\mathbf{x}^j|\alpha^j, \Phi) P(\alpha^j) d\alpha^j. \tag{6}$$

The analytic solution of this integration is in general intractable. Nevertheless, since α^j is expected to have a tightly compact and sparse distribution, the integral can be approximated by evaluating $P(\mathbf{x}^j|\Phi)$ at its maximum. Then, we get

$$\Phi^* = \arg\max_{\Phi} \sum_{\forall j} \max_{\alpha^j} \left\{ \log\left(P(\mathbf{x}^j|\alpha^j, \Phi) P(\alpha^j) \right) \right\}. \tag{7}$$

For performing the optimization, the probabilities need to be defined. Thus, by assuming the noise term in Equation (1) to be Gaussian i.i.d, we have

$$P(\mathbf{x}^j|\alpha^j, \Phi) = \frac{1}{Z_x} \exp\left(-\frac{1}{2\sigma^2} ||\mathbf{x}^j - \Phi\alpha^j||^2 \right), \tag{8}$$

where Z_x is a normalization constant and $\sigma^2 I$ is the covariance matrix of i.i.d Gaussian noise ν. Moreover, to enforce sparsity, the prior distribution of the coefficients, $P(\alpha^j)$, is assumed to be Laplacian or Cauchy:

$$P(\alpha) = \frac{1}{Z_c} \exp\left(-\lambda_1 ||\alpha^j||_1 \right), \tag{9}$$

where Z_c is a normalization constant and $||.||_1$ is $\mathcal{L}_1$ norm. This model ensures that most of the entries of α^j will be zero. Integrating Equations (8)-(9) into Equation (7), the overall problem becomes

$$\Phi^* = \arg\min_{\Phi} \sum_{\forall j} \min_{\alpha} \left\{ \frac{1}{2\sigma^2} ||\mathbf{x}^j - \Phi\alpha^j||^2 + \lambda_1 ||\alpha^j||_1 \right\}. \tag{10}$$

By imposing different constraints on basis Φ, many iterative solutions are proposed to solve the problem [51, 52, 44]. Moreover, by considering different priors on $P(\Phi)$

(such as unit Frobenius or $\mathcal{L}_2$ norm,) the maximum-a-posteriori estimation approach designed in [40] provides improvements over ML-based models.

The most important achievement of this probabilistic framework for dictionary design is the iterative optimization scheme, which first sparsifies the coefficients and then updates the dictionary elements. This approach later give way to successful dictionary learning methods [31, 43, 2]. In the next section, the review is geared toward image compression and transform coding.

2.3 Transform Coding

Transform coding has long been the standard data compression method that produces state-of-the-art results. The fundamental idea in transform coding is to exploit the regularity within data samples such that the redundancy of the representation is minimized subject to a fidelity cost. Basically, transforms enable signal decorrelation, which packs the signal energy into only a few coefficient. Encoding these coefficients rather than the original signal provides significant bitrate savings. The main target of codec design, therefore, is to sustain a given level of distortion between the original and the reconstructed signal while reducing the number of bits spent for encoding the data (i.e., bitrate).

Figure 1 shows a flow chart of transform-based codec. The system is divided into three main blocks: encoder of the transmitter, decoder at the receiver, and the channel. In our design the output of the encoder is assumed to be the same as the input of the decoder, hence the channel is lossless. In the codec, first the input data, x, is transformed to have sparse representation. Next, the continuous range of the coefficient values of the transform is constrained to have discrete levels in the quantization process. With the quantization, one can expect to have the most of the coefficients to be zero. Finally, the coefficient levels are entropy coded by Huffman or arithmetic coding. This basic flow is reversed in the decoder to recover the original

signal or its approximations.

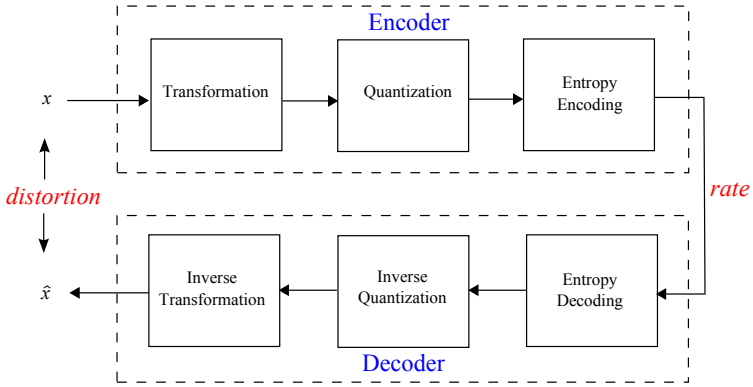


Figure 1: Flow chart of a transform-based codec.

2.3.1 Orthonormal versus Overcomplete transforms

For data compression, the orthogonal transforms are often chosen due to their decorrelation properties and ease of their implementations. In this and the next chapters, our main focus is on designing orthogonal or bi-orthogonal transforms to improve coding efficiency of the existing methods. The choice of having orthogonal transforms rather than overcomplete ones for compression can be justified by the design requirements of the codec quantizers. Basically, with orthogonal transforms, the quantization of each coefficient have direct and independent effect on the amount of distortion introduced to the reconstructed signal, due to Parseval's theorem. This also means, when the coefficients of an orthogonal transform is quantized to the nearest level, it will have minimal effect to the overall signal reconstruction. Whereas for overcomplete representation, since the atoms of the transform are not independent, a search algorithm is needed to find the best quantization levels that will give the minimum distortion.

To explain this, Figure 2-(a) depicts a 2D orthogonal system and Figure 2-(b) shows two basis vectors of an overcomplete system in 2D, where the corresponding quantization levels are the circles. For the orthogonal setting in Figure 2-(a), the observed signal (star) is quantized to nearest circle, which also guarantees minimum

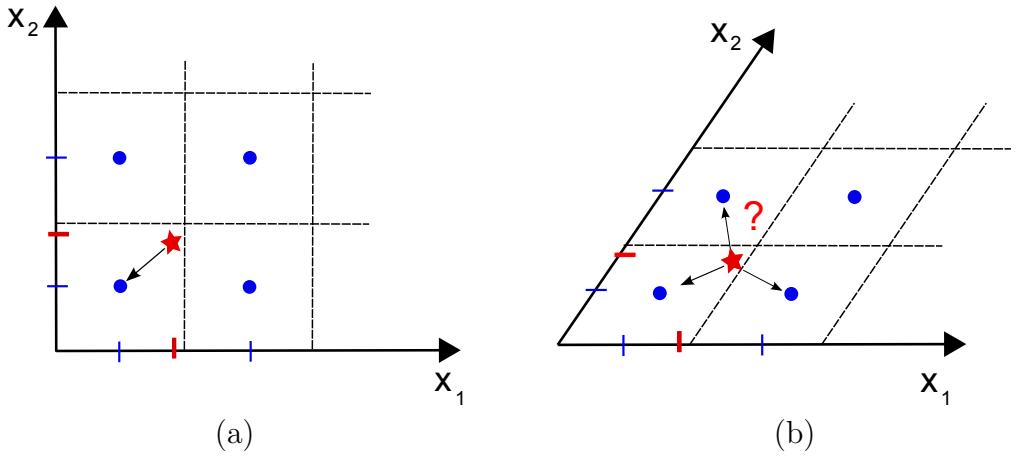


Figure 2: Quantization constellation for orthogonal (a), and overcomplete systems (b). Here stars represent a real valued signal vectors and circles are geometric centers of quantization intervals.

distortion. With the same quantization levels and coefficient values, a search algorithm is need for overcomplete systems to locate the quantization levels that will give minimum distortion.

The advantage of having an overcomplete representation stems from the structural diversity of the basis vectors, which can provide sparse signal representation with appropriate solvers, such as $\mathcal{L}_1$ regularized minimizers (LASSO etc.). Although these solvers are becoming faster, they are still not practical for general purpose image/video codec design. The implementation for orthonormal transforms, on the other hand, is straight forward matrix multiplication. One way to improve the structural diversity of the orthonormal representations is to have a transform adaptation methodology, which seeks for a sparser representation among multiple candidate transforms. Our endeavor is to increase the representation efficiency of transforms by designing a general framework, which first optimizes a set of transforms over some data then selects these transforms adaptively to compress the data, all in accordance with the rate-distortion analysis.

One can group the orthonormal (or bi-orthonormal) transforms into three main

categories, depending on their structures: 1) block transforms [3], 2) lapped transforms [50, 49], and 3) wavelet transforms [16, 4]. Each structure has advantages in different applications. For example, block and lapped transforms are preferred in video and image coding due to low complexity and fewer memory access requirements. On the other hand, for smooth images with point singularities, it is hard to compete with the rate-distortion performance of wavelet transforms. Nevertheless, regardless of transform structure, the compression efficiency of transform coders decreases significantly on directional image singularities [46].

Among block transforms, the discrete cosine transform (DCT) [3] became the industry standard for video and image coding applications by offering near-optimal compression performance with fast evaluation algorithms [56, 75]. Although the Karhunen-Loeve transform (KLT) for linear approximation provides a statistically optimal representation in a mean-square error sense, it is not preferred because of its signal-dependent nature. Recently, the directional transforms generated by KLT started to appear in video coding applications [78] to increase coding efficiency. Since KLTs are signal dependent, the method in [78] uses the prediction directions of H.264's intra-coding for its basis adaptation.

Although block transforms provide visually lossless reconstruction at high bitrates, the blocking and the ringing artifacts around the transform boundaries become visible as the bitrate decreases. In Figure 3-a, the reconstruction result of a DCT-based codec is given at 0.15 bits per pixel (bpp). Note that the blocking artifacts around edges and textured regions are visible. This artifacts occurs because the mean values of the reconstructed blocks are changed independently after the quantization. To remove the blocking artifacts, Malvar designed lapped transforms, which borrow pixels from adjacent blocks such that the coefficients of the adjacent blocks become dependent such that the effect of quantization on the block means is kept around the same

(a) (b)

Figure 3: Reconstructed foreman image at 0.15bpp with, (a) DCT, and (b) LBT based image codecs.

[50, 49]. Without any increase in the number of coefficients, lapped transforms provide better energy compaction and reduce the blocking artifacts compared to DCT. Figure 3-b shows the reduction of blocking artifacts in the reconstructed image when the lapped bi-orthonormal transform (LBT) is used [50]. Although the lapped transforms have superior performance over the DCT-based codecs, they have not yet become part of standardization efforts, which is partly because of their increased complexity. Similar to the block transforms, the coding performance of the lapped transform can be improved by adding directional features into representation [1, 63] with appropriate basis adaptation.

Different than the block and the lapped transforms, the wavelet representation offers multiresolutional approximation of a given signal. The entire characterization of the signal is governed by a discrete set of filters that controls information transfer across resolutions. Basically, a signal is approximated by dilated and shifted versions of a scaling function that forms orthonormal or bi-orthonormal wavelet basis functions. Projecting the signal onto these wavelet bases enables very efficient transform-domain representation, which is used in state-of-the-art image coders [71, 64, 60].

Nevertheless, the compression performance of wavelet transforms suffers around directional edges like block and lapped transforms. The available literature related to improving the shortcomings of the wavelet transforms at directional singularities can be grouped into two mainstream approaches. The first category of work formulates 2D geometrical singularities in the signal domain and develops representations that provide sparse decompositions over them. Curvelets [12], contourlets [22], and first-generation bandelets [57] can be grouped into this class. The directional lifting-based methods such as those in [34, 14, 21] can also be included in this group since they modify the prediction step of the original filter-bank into a directional prediction step. The second category, on the other hand, formulates the problem in the transform domain and proposes representations in terms of transform coefficients. Wavelet-footprints [28], wedge-prints[74], and second generation bandelets [58] can be counted in this group. Foot-prints and wedge-prints exploit inter-scale dependencies over singularities by introducing a vector dictionary, assuming step edges. Second-generation bandelets adaptively reorder coefficients and apply a secondary wavelet transform. In general, these methods improve decorrelation over singularities with the aid of directional side information referred to as geometric flow.

The sparse representation achieved by the wavelet decomposition ensures that the wavelet coefficients of a smooth region will be rapidly decaying. Since wavelet representation enables localization in space and (spatial) frequency, the coefficients of a region at different resolutions can be grouped into trees. Hence, many of the coefficients of a tree in smooth regions are likely to have small (or insignificant) magnitudes. On the other hand, if the region contains a singularity, the wavelet coefficients within corresponding trees are expected to be significant. The wavelet-based compression algorithms such as those in [60, 64] utilize these observations and obtain significant gains whenever the number of significant trees is small.

In a more analytical setting, using continuous-time analysis, one can show that the

distortion due to the compression of uniformly smooth signals with point singularities asymptotically complies with

$$D(R) \lesssim R^{-\alpha}, \tag{11}$$

where R is the number of bits spent to represent the signal and α quantifies the local smoothness of the signal, with larger α corresponding to smoother signals [17, 46]. Note that regardless of the point singularities, the operational distortion-rate function tracks the smoothness of the signal and obtains the asymptotically optimal performance (see [17] for conditions). In comparison, when the signal contains singularities along curves, one obtains

$$D(R) \lesssim R^{-1}, \tag{12}$$

regardless of how large α is. Hence, isolating singularities and using better decorrelation techniques has become a focal point of recent image compression related research.

CHAPTER III

SPARSE ORTHONORMAL TRANSFORMS

The method we introduce in this chapter is a very generic, algebraic transform optimization that can be used to exploit the regularity along the directional edges to increase the coding efficiency of transforms. To show the flexibility of the proposed method in application to different data types, the transform optimizations are done at both the signal and transform domains. We introduce our method by optimizing block transforms in signal-domain. As an extension of the proposed technique, the optimization of lapped and wavelet transforms is done with transform-domain data [63, 61]. Our construction differentiates itself from the rest of the literature, which mainly focuses on model-based transform design, by offering a data-driven optimization of the transform representations. Moreover, although our main focus will be on image data, the proposed method is data agnostic and can be extended to optimize transforms for audio, speech, or graphics data.

Built around the core sparsity ideas, blocks of data are classified based on their geometry, and then conditionally sparse transforms are designed for each class. For this purpose, a large number of blocks of the same size are initially collected from natural images and heuristically grouped into k different classes. This heuristic can be a K-MEANS algorithm or any other clustering method that can surface the structural differences within the signal database. After designing optimally sparse transforms for the initial classification of each class, we reclassify the blocks and repeat the process until the classification and associated transforms are jointly optimal. The cost function we use in the optimization of classifications and for the transform design is based

on $\mathcal{L}_0$ norm regularization. This effectively replaces the analytically intractable rate-distortion optimization with non-linear approximation based optimization. This is justified because nonlinear approximation with optimized transforms behaves asymptotically similar to rate-distortion optimized results [17]. The end result of our process gives us k classes and k orthonormal transforms. For image compression, which is the main focus in this work, the initial heuristic that we use to classify blocks is based on directional structure of the blocks. Surprisingly this structure is preserved at the end of the joint optimization, which is in line with results obtained by other researchers [7, 52].

This optimization is extended to improve the transform-domain representation for lapped and wavelet transforms. For this, we design sparsity-distortion optimized orthonormal transforms that exploit the inter-scale and intra-scale dependencies of transform coefficients. The new orthonormal transforms, when applied on the top of a given lapped or wavelet transform, map their coefficients over signal singularities to a sparser representation.

3.1 Construction of Sparse Orthonormal Transforms (SOT)

Rate-distortion optimized transform design aims to find the best orthonormal transform(s) that will minimize the distortion level for a given bit budget (B). Let Φ be an orthonormal transform and assume image block $\mathbf{x} \in \mathcal{S}$,where $\mathcal{S}$ is the set of image blocks that are extracted from natural images. As a general transform optimization we can write

$$\min_{\Phi} \mathbb{E}\left[\mathcal{D}(\Phi;\mathbf{x})\right] \quad s.t. \quad \mathbb{E}\left[\mathcal{R}(\Phi;\mathbf{x})\right] \leq B, \tag{13}$$

where $\mathcal{D}(\Phi;\mathbf{x})$ and $\mathcal{R}(\Phi;\mathbf{x})$ are distortion and rate of compressing image block $\mathbf{x}$ with transform Φ and bit budget B. The expectations $\mathbb{E}\left[\mathcal{D}(\Phi;\mathbf{x})\right]$ and $\mathbb{E}\left[\mathcal{R}(\Phi;\mathbf{x})\right]$ are obtained over $\mathcal{S}$. Using a Lagrange multiplier λ, the above problem can be

reformulated into an unconstrained minimization as

$$\min_{\Phi} \mathbb{E}\left[\mathcal{D}(\Phi;\mathbf{x}) + \lambda\mathcal{R}(\Phi;\mathbf{x})\right]. \tag{14}$$

From this most general formulation, our construction focuses on structural differences of signal singularities; hence, clusters signal singularities into classes to find the optimal orthonormal transform that will minimize the expected rate-distortion cost of each class $k \in \{1, ..., K\}$:

$$\min_{\Phi_k} \mathbb{E}\left[\mathcal{D}(\Phi_k;\mathbf{x}_k) + \lambda\mathcal{R}(\Phi_k;\mathbf{x}_k)\right] \tag{15}$$
$$s.t \quad \Phi_k^T\Phi_k = \mathbf{I}$$

where $\mathbf{x}_k$ is a block of a signal with type k singularity, K is the total number of singularity classes, and Φ_k denotes the transform for the k^{th} class. Various techniques can be used to generate these structural classes. The one employed in this paper groups blocks with respect to their geometric structure [57]. Surprisingly this structure is preserved at the end of the joint optimization of the transforms and the classes, which is in line with results obtained by other researchers [7, 52].

Two aspects of Equation (15), the rate term and the expectation, are further specialized for the proposed Sparse Orthonormal Transforms (SOT). Since determining an analytical expression for rate at a given distortion level is difficult, we approximate the rate to arrive at a more tractable cost function. We resolve expectations by using a training set. Let S_k be a set of training signal blocks of type k. The simplified form of (15) becomes

$$k = \{1, ..., K\}:$$
$$\min_{\Phi_k} \left\{ \sum_{\mathbf{x}^j \in \mathcal{S}_k} \min_{\alpha_k^j} \|\mathbf{x}^j - \Phi_k\alpha_k^j\|_2^2 + \lambda\|\alpha_k^j\|_0 \right\} \tag{16}$$
$$s.t \ \Phi_k^T\Phi_k = \mathbf{I}$$

where $\mathbf{x}^j$ is the j'th block in $\mathcal{S}_k$ lexicographically ordered into a vector and α_k^j denotes the transform coefficients of $\mathbf{x}^j$ after application of Φ_k. Here, the rate is estimated through nonlinear approximation ($\mathcal{L}_0$ norm,$\|.\|_0$) by counting the number of nonzero coefficients and the Euclidean norm is used to estimate distortion (refer to [47, 17] for justifications.)

Using iterative conditional minimization over coefficients and transforms then reclassifying the training set, one can reach a set of optimized orthonormal transforms as suggested in [63]. The solution to Equation (16) has two basic steps:

3.1.0.1 Transform Optimization

In the iterative transform optimization process, all variables are assumed to be fixed except the one that is optimized over the cost function. There are three variables in our formulation; transform matrix, coefficient vector, and Lagrange multiplier. With the given initial transforms, the optimization first updates the coefficients. Next, the transforms are changed according to the updated coefficient values. For the Lagrange multiplier, a fixed value can be assumed throughout the transform optimization step. Yet, we have designed an annealing process, which sweeps a range of Lagrange multiplier values so that transforms are not tuned to a single λ value.

Optimal coefficients for a given transform

This step imposes sparseness over the coefficients of a given orthonormal transform Φ_k. The initial transform can be KLT of the members of the class or just fixed DCT. The imposed sparseness has to be balance with the distortion, thus new coefficients are found by

$$\alpha_k^j = \arg\min_{\mathbf{d}} (\|\mathbf{x}^j - \Phi_k \mathbf{d}\|_2^2 + \lambda \|\mathbf{d}\|_0) \quad (17)$$
$$s.t. \quad \Phi_k^T \Phi_k = \mathbf{I}.$$

The solution to this equation is hard-thresholding, in which the components of $\mathbf{d} = \Phi_k^T \mathbf{x}^j$ that are smaller than the threshold $\sqrt{\lambda}$ is set to zero:

$$\alpha_k^j(l) = \begin{cases} \mathbf{d}(l) & ; \quad |\mathbf{d}(l)| \geq \sqrt{\lambda} \\ 0 & ; \quad |\mathbf{d}(l)| < \sqrt{\lambda} \end{cases}, \quad 1 \leq l \leq N.$$

Optimal transforms for given coefficients

The next step for transform optimization is to determine the optimal orthonormal transforms that will minimize the reconstruction error for the updated coefficients. This optimization problem is also known as orthogonal procrustes [**?**], and here we present a novel algebraic solution to it. Following the Equation (16), the new minimization can be formulated as:

$$\Phi_k = \arg\min_{\Psi} \left(\sum_{\mathbf{x}^j \in \mathcal{S}_k} \|\mathbf{x}^j - \Psi \alpha_k^j\|_2^2 \right) \tag{18}$$

$$s.t. \quad \Psi^T \Psi = \mathbf{I}. \tag{19}$$

Note that the $\mathcal{L}_0$-norm term vanishes, since the coefficients are fixed. Using the relation of $\mathcal{L}_2$ norm with the trace of matrices, we can rewrite the above equation as:

$$\min_{\Psi} \sum_{\mathbf{x}^j \in \mathcal{S}_k} [Tr((\mathbf{x}^j - \Psi \alpha_k^j)^T (\mathbf{x}^j - \Psi \alpha_k^j))] \tag{20}$$

$$s.t. \quad \Psi^T \Psi = \mathbf{I}.$$

where Tr denotes the trace of a matrix. Rearranging terms results in

$$\min_{\Psi} \sum_{\mathbf{x}^j \in \mathcal{S}_k} [-2Tr(\alpha_k^j \mathbf{x}^{jT} \Psi)]. \tag{21}$$

Let $\mathbf{Y} = \sum_{\mathbf{x}^j \in \mathcal{S}_k} \alpha_k^j \mathbf{x}^{jT}$ and let $\mathbf{U}\Lambda^{1/2}\mathbf{V}^T$ denote the SVD of $\mathbf{Y}$ [68]. Note that $\mathbf{U}$ and $\mathbf{V}$ are orthonormal and Λ is diagonal. Equation(21) becomes

$$\max_{\Psi} [Tr(\mathbf{Y}\Psi)] \;=\; \max_{\Psi} [Tr(\mathbf{U}\Lambda^{1/2}\mathbf{V}^T\Psi)], \tag{22}$$

and the optimization formula can be written as

$$\max_{\Psi}[Tr(\Lambda^{1/2}\mathbf{V}^T\Psi\mathbf{U})] \qquad s.t. \quad \Psi^T\Psi = \mathbf{I}. \tag{23}$$

Let $\mathbf{P} = \mathbf{V}^T\Psi\mathbf{U}$, since $\mathbf{V}$, Ψ, and $\mathbf{U}$ are orthonormal, so is $\mathbf{P}$. Then we have

$$\max_{\mathbf{P}}[Tr(\Lambda^{1/2}\mathbf{P})] \qquad s.t. \quad \mathbf{P}^T\mathbf{P} = \mathbf{I}. \tag{24}$$

Since Λ is a diagonal matrix,

$$Tr(\Lambda^{1/2}\mathbf{P}) = \sum_{l=1}^{N} \Lambda^{1/2}(l,l)\mathbf{P}(l,l). \tag{25}$$

By definition of SVD, Λ has non-negative entries. Also, using $\mathbf{P}^T\mathbf{P} = \mathbf{I}$ and $|\mathbf{P}(l,l)| \leq 1$, the Equation (24) is maximized when $\mathbf{P} = \mathbf{I}$. Thus, the Equation (20) is minimized by

$$\mathbf{V}^T\Phi_k\mathbf{U} = \mathbf{I} \quad \Rightarrow \quad \Phi_k = \mathbf{V}\mathbf{U}^T. \tag{26}$$

The expression in Equation (16) can thus be solved by successive minimization of the cost functions shown in Equations (17) and (18) until the cost reaches a steady state. The value of Lagrange multiplier, λ is a parameter that can be experimentally determined for a particular set of signals. Or an annealing step can be added to transform optimization.

Annealing λ

The Lagrange multiplier, λ, sets the balance between the distortion and the sparsity (or rate) expressions. Since λ is proportional with the square of the quantization level, a large value for λ tunes the transforms to high distortions or low bit-rates. Rather using a fixed λ, a more general transform representation can be achieved by annealing. For this process, starting from a large value, λ is deterministically lowered to zero. This effectively reduces the influence of rate in the minimization. Experimentally such annealing schemes is shown to provide results closer to global minimum [10], which will be confirmed in our compression results as well.

Figure 4 shows $\sqrt{\lambda}$ value at which the basis vectors converge during the annealing process. In general, at high lambda values, shown with brighter colors in Figure 4-(a), the components have lower frequencies in the horizontal direction. As λ is decreased, the components with higher horizontal frequencies are observed to reach steady state. The order of components in Figure 4-(b) is the result of the proposed algebraic optimization. For efficient compression, this order would be changed in the experiments to better utilize the entropy coder.

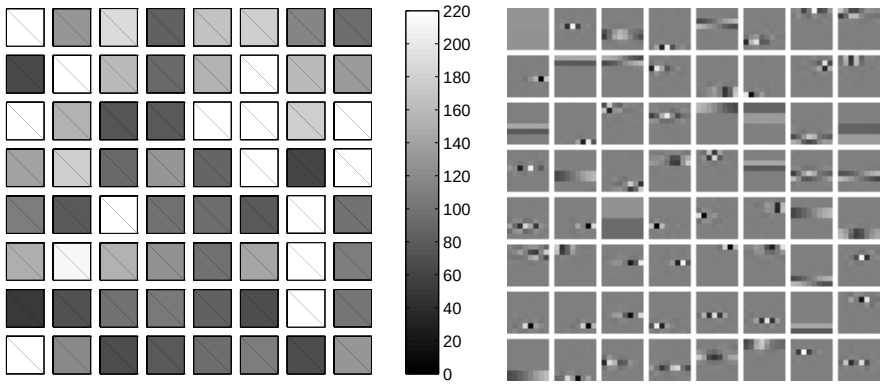


Figure 4: A one-to-one color mapping of $\sqrt{\lambda}$ at which corresponding basis vector in (b) converges to a steady state in the optimization, (a). Sparse orthonormal transform designed for horizontal direction, (b).

3.1.0.2 Reclassification

After the transform optimization is completed for all classes, each block is assigned to a new class that provides minimal sparsity-distortion cost. This way, the optimization of the members of the classes and their corresponding orthonormal transforms are coupled. The reclassification is done as follows:

$$Label\{\mathbf{x}^j\} = \arg\min_k [\min_{\alpha_k^j} [\|\mathbf{x}^j - \Phi_k \alpha_k^j\|_2^2 + \lambda \|\alpha_k^j\|_0]], \tag{27}$$

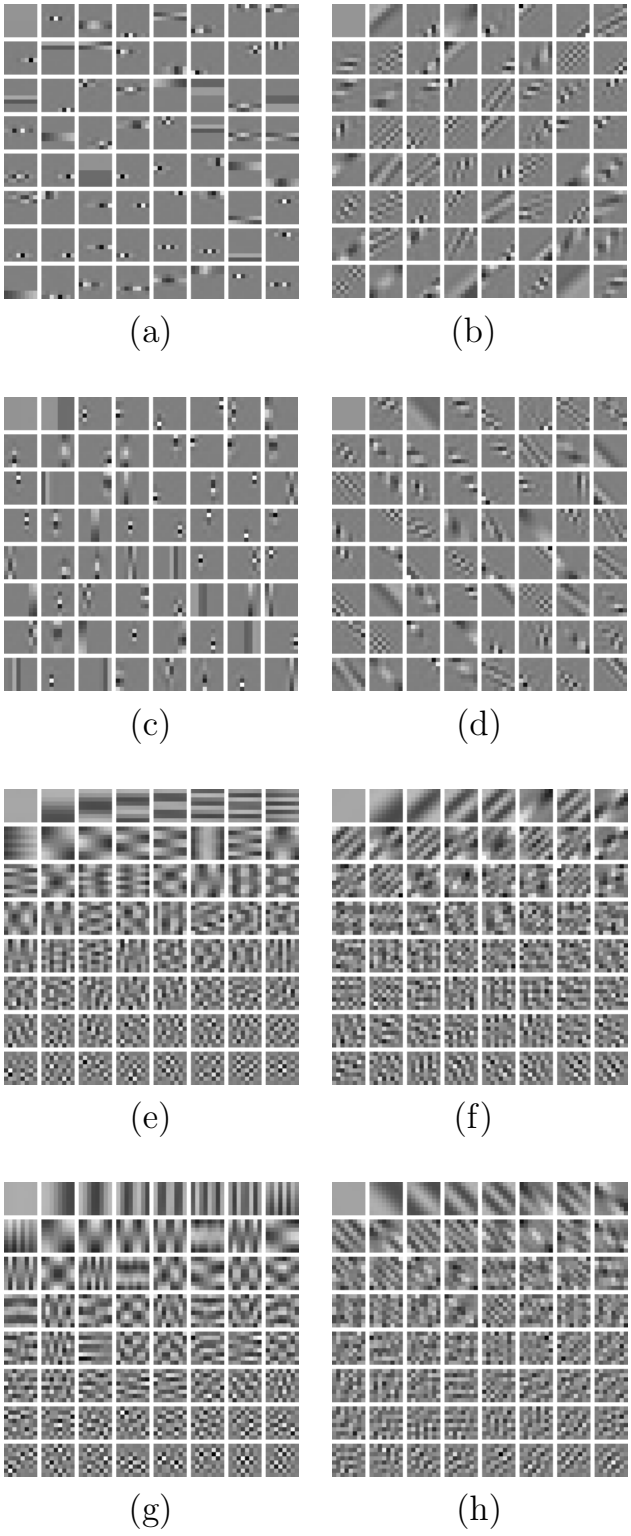


Figure 5: Sparse orthonormal transforms aligned with, (a) 0 degree, (b) 45 degree, (c) 90 degree, and (d) 135 degree image gradients. Karhunen-Loeve transforms (KLT) of same training data for, (e) 0 degree, (f) 45 degree, (g) 90 degree, and (h) 135 degree image gradients

which furnishes new $\mathcal{S}_k$'s. The iterations of transform optimization and reclassification steps continue until the sparsity-distortion cost of the system converges. The details of the iterative optimization are given in Algorithm 1. Figure 5 shows block transforms of four different classes optimized by the proposed method and the corresponding transforms of Karhunen-Loeve's method (KLT) for those classes.

3.1.1 Its Relation with K-Means

The proposed optimization method provides us a dictionary learning method for clustering and classification, as well. In K-Means clustering, a set of centroids that provides best fit is sought after and class label of each element is determined as follows:

$$Label\{x\} = \arg\min_k \|x - c_k\|_2^2 \tag{28}$$

where c_k is the centroid of the k-th class. The proposed optimization method, on the other hand, finds a set of dictionaries that signal is best reconstructed. Later, the decision on which class an element belongs to is made by

$$Label\{x\} = \arg\min_k \big(\min_\alpha \|x - \Phi_k \alpha\|_2^2 + \lambda \|\alpha\|_0 \big). \tag{29}$$

One can say that the described algorithm works like K-Means, which both do hard assignments. We would like to refer interested readers to the work of Dremeau and Herze [29], in which a probabilistic framework built on Expectation-Maximization (EM) is proposed for the coupled designed of structural classes and their corresponding transforms, in line with our deterministic approach.

3.2 Sparse Lapped Transforms (SLT)

Lapped transforms are proposed as an efficient way to reduce the compression artifacts of block-based transforms at low bit rates [50, 49]. The lapped transforms borrow samples from neighboring blocks, yet the number of coefficients is same as the number of samples (hence it is critical sampling). The extension of the described formulation

Algorithm 1: Iterative Optimization for Sparse Orthonormal Transforms

Data: A training set of blocks $\mathcal{S}$ and initial transforms that will be optimized $\Phi_k^{(0)}$'s
Result: Sparsity-distortion optimized set of orthonormal transforms Φ_k's and corresponding classes $\mathcal{S}_k$'s.

0- Initialization:

1 Partition training set $\mathcal{S}$ into K different sub-classes, $\mathcal{S}_k$, with respect to image gradients

2 Set Φ_k equal to DCT

I- Basis Update: $\forall k = \{1, ..., K\}$, set λ large;

1 Find optimal coefficients for given transform Φ_k, for all $\mathbf{x}^j \in \mathcal{S}_k$;

Compute $\alpha_k^j = \arg\min_{\mathbf{d}}(\|\mathbf{x}^j - \Phi_k \mathbf{d}\|_2^2 + \lambda \|\mathbf{d}\|_0)$ by hard-thresholding
Solution is:

$$\alpha_k^j(l) = \begin{cases} \mathbf{d}(l) & ; \quad |\mathbf{d}(l)| \geq \sqrt{\lambda} \\ 0 & ; \quad |\mathbf{d}(l)| < \sqrt{\lambda} \end{cases}, 1 \leq l \leq N$$

where $\mathbf{d} = \Phi_k^T \mathbf{x}^j$.

2 Find optimal orthonormal transform for all α_k and $\mathbf{x}^j \in \mathcal{S}_k$;

Compute $\Phi_k = \arg\min_{\Psi} \left(\sum_{\mathbf{x}^j \in \mathcal{S}_k} \|\mathbf{x}^j - \Psi \alpha_k^j\|_2^2 \right) \quad s.t. \quad \Psi^T \Psi = \mathbf{I},$

Solution is: $\Phi_k = \mathbf{V}\mathbf{U}^T$

where singular value decomposition of $\sum_{\mathbf{x}^j \in \mathcal{S}_k} \alpha_k^j \mathbf{x}^{jT}$ is $\mathbf{U}\Lambda^{1/2}\mathbf{V}^T$

3 Check for convergence.

If convergence is reached; Reduce λ
If $\lambda > 0$; Go to step 1.
Else RETURN Φ_k.
Else Go to step 1.

II- Reclassification:

Relabel all block $\mathbf{x}^j \in \mathcal{S}$ with the label of orthonormal transforms minimizing the cost;

$Label\{\mathbf{x}^j\} = \arg\min_k [\min_{\alpha_k^j} [\|\mathbf{x}^j - \Phi_k \alpha_k^j\|_2^2 + \lambda \|\alpha_k^j\|_0]].$

RETURN $\mathcal{S}_k$'s.

III- Overall Convergence Check:

If the sparsity-distortion cost converges; RETURN Φ_k's and $\mathcal{S}_k$'s EXIT.

Else Go to Step I.

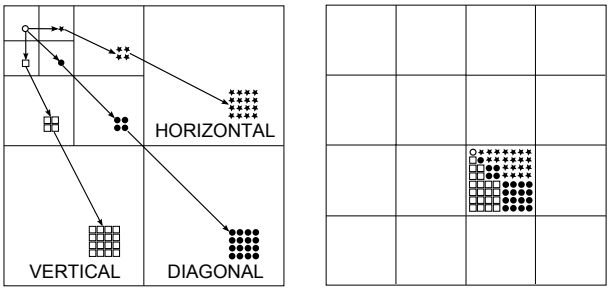
to lapped transforms is straightforward. We start with an initial transform such as lapped orthogonal transforms (LOT) or lapped bi-orthogonal transforms (LBT) [50].

Let $\mathbf{A}_F(N \times M), \mathbf{A}_I(M \times N), M > N$ denote the initial lapped transform and its inverse. Let $\mathbf{y}^j(M \times 1)$ denote the j-th spatial block used for training. In the lapped formulation, the same iterative optimization process is performed as discussed. However, the training data is different. In the lapped case, the training blocks $\mathbf{x}^j(N \times 1)$ correspond to the lapped transform coefficients: $\mathbf{x}^j = \mathbf{A}_F\mathbf{y}^j$. Then, same method is used for computing $\Phi_k(N \times N)$, and the final sparse lapped transforms are $\Phi_k^T\mathbf{A}_F$, and the inverse are $\mathbf{A}_I\Phi_k, (k = 1, \ldots, K)$. Similar to sparse lapped transform (SLT) optimization described above, it is also possible to design transforms over DCT coefficients. If optimization is done right, the combination of such a transform and DCT will give results similar to SOT (since the multiplication of two orthonormal transforms is another orthonormal transform); hence further discussions are omitted.

The SLT optimization has very appealing simplicity in directional lapped transform design. Note that without using complex directional modulation properties of lapped transforms, SLTs will possess directional features that increases its coding efficiency across edges.

3.3 Sparse Multiresolutional Transforms (SMT)

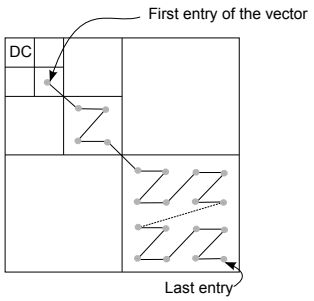
The extension of the proposed optimization method to wavelet transform is also formulated in the transform domain. The major difference between SLT and SMT is the way coherence of the coefficients is utilized. In SLT, we directly mapped the coefficients of the original lapped transform into a new transform space. Wavelet transforms, on the other hand, require a different approach in which the coherence of the wavelet coefficients at different subbands is exploited. Consider a multiscale decomposition of an image with a 2D discrete wavelet transform. Using the idea in [76], one can form $n \times n$ blocks of wavelet coefficients from the original wavelet



(a) (b)

Figure 6: Subbands of a three-level discrete wavelet transform in (a) is mapped to blocks of wavelet coefficients in (b)

decomposition of the entire image as shown in Figure 6. Beyond the lowest frequency subband, such a decomposition represents an image in terms of subbands having three different orientations: vertical, diagonal, and horizontal subbands (refer to Figure 6-(a) for the subband orientations). Since the wavelet coefficients of a region have strong coherence among the subbands with same orientation, we have defined vectors of wavelet coefficients for each subband orientation. For the blocks of wavelet coefficients in Figure 6-(b), Figure 7 shows how a sub-tree of wavelet coefficients in the diagonal subbands is ordered into a vector. Depending on the level of wavelet decomposition, the size of this vector, hence the transform applied on to it, changes.



(a)

Figure 7: A vector of coefficients for diagonal subbands is extracted by the given scanning order.

In our new wavelet decomposition method, a set of orthonormal transforms is

optimized to each subband orientation. The overall optimization then becomes

$$k \in \{1, ..., K\}, o \in \{V, D, H\} :$$
$$\min_{\Phi_k^o} \left\{ \sum_{\mathbf{s}_i \in \mathcal{S}_{(k,o)}} \min_{\alpha_i} \|\mathbf{s}_i - \Phi_k^o \alpha_i\|_2^2 + \lambda \|\alpha_i\|_0 \right\} \tag{30}$$
$$s.t \ \Phi_k^{oT} \Phi_k^o = \mathbf{I}$$

where $\mathbf{s}_i$ is the i'th sub-tree of class k and subband orientation o in the training set $\mathcal{S}_{(k,o)}$, which is lexicographically ordered into a vector as shown in Figure 7. Here, α_i denotes the transform coefficients of $\mathbf{s}_i$ with Φ_k^o. Using the iterative conditional minimization given in section 3.1, it is possible to generate a new set of orthonormal transforms, which are named as sparse multiresolutional transforms (SMT).

3.4 Experiments/Simulations

The proposed optimization can be applied to various signal processing or restoration problems. In this chapter, three prototype image (with block, lapped, and wavelet transform structures) codecs are designed to evaluate the compression performance of the proposed approach.

3.4.1 Dictionary Learning

The experiments start with dictionary learning step which is performed offline. For this, we used around 50,000 blocks and their transforms selected from a set of natural images. Standard block size is set to 8×8. To extract lapped transform coefficients of a block, we used LBT [50]. Our choice for wavelet transform is a 3-level CDF-9/7 wavelet transform (similar to JPEG2000.) Three levels of the wavelet coefficients with the same subband orientation are grouped into sub-trees as shown in Figure 7. Such a sub-tree corresponds to a region of size 8×8 in signal domain. The training images were not used in the simulation results we report. As the initial heuristic, the blocks are classified based on the image gradient, which varies from 0 to 157.5 with 22.5

degree intervals. This process results in 8 classes. Our optimization method is not designed to preserve the gradient based classification, however, the final optimized basis functions do have directional structure as illustrated in Figure 5. In addition to these directional classes, we include the DCT in the rate-distortion optimization stage for block-transform image codec and an identity matrix for lapped- and wavelet-transform codecs, which results in a total of 9 classes and transforms for each codec.

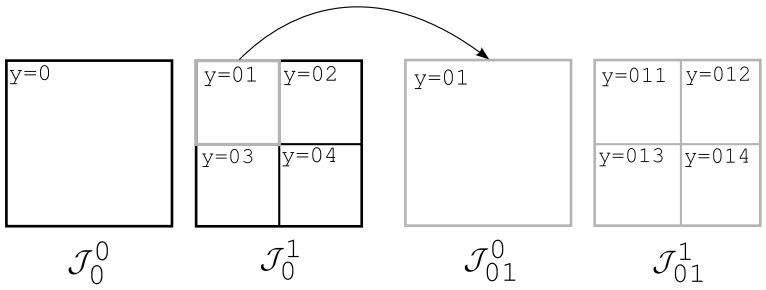


Figure 8: Quadtree segmentation. Labels of segments are in the top-left corners.The abbreviations for sparsity-distortion cost of an encoding unit are given at the bottom.

3.4.2 Transform Adaptation

Having a library of orthonormal transforms requires appropriate adaption to the structure of data. The proposed method adapts transforms in a sparsity-distortion optimal fashion similar to a CART-like algorithm [57]. The Figure 8 shows the segmentation of an encoding unit with labels for the costs and the segments. For a non-partitioned (or a leaf node) segment the encoding cost is determined as follow

$$\mathcal{J}_{\mathsf{y}}^{\mathsf{0}} = \min_{k} \left(\sum_{\forall \mathbf{x} \in \mathcal{Q}_{\mathsf{y}}} \min_{\alpha} \|\mathbf{x} - \Phi_k \alpha\|_2^2 + \lambda \|\alpha\|_0 \right) + E^{\mathsf{0}} \tag{31}$$

where $\mathcal{Q}_{\mathsf{y}}$ includes all non-overlapping 8×8 blocks in the segment y. E^{0} is the cost of signaling the class information, which is proportional to the number of bits spend to encode the classes of the transforms. To check if the segment y is a leaf node or not, y will be segmented into four parts, and following cost is calculated

$$\mathcal{J}_{\mathsf{y}}^{\mathsf{1}} = \sum_{j=1}^{4} \min(\mathcal{J}_{\mathsf{y}_j}^{\mathsf{1}}, \mathcal{J}_{\mathsf{y}_j}^{\mathsf{0}}) + E^{\mathsf{1}}. \tag{32}$$

For $\mathtt{y}$ to be a leaf node, these two cost values of the segment $\mathtt{y}$ should obey $\mathcal{J}_{\mathtt{y}}^{\mathtt{1}} > \mathcal{J}_{\mathtt{y}}^{\mathtt{0}}$. Here $E^{\mathtt{1}}$ is the segmentation cost. A simplified flow-chart of the quadtree segmentation algorithm is provided in Figure 9.

Figure 10 shows the quadtree partitioning of the described transform adaptation method with two λ values. Here the arrows represent the class of the transform that is used to encode all 8×8 blocks with that particular segment. Observe that as the λ value gets larger the geometric adaptation of the transforms gets coarser, i.e., larger segments are generated. It is important to note that this structure and the fact that the structure is geometric is discovered by the classification resulting of an algebraic optimization process.

3.4.3 Image Codecs

The proposed block-transform codec divides image into 8×8 non-overlapping blocks and finds coefficients of each block with the transform provided by the quadtree segmentation algorithm. For lapped- and wavelet-transform codecs, the original coefficients are replaced with the coefficients of new transforms. An important issue for entropy coding is the order of transforms. State-of-the-art entropy coders utilize a priority scheme for significant and insignificant coefficients. If the transform coefficients are not placed in correct order before entropy coder, the rate will increase drastically. Since the significance of a coefficient is related with its energy, we proposed an energy-based ordering of coefficients. First, the vectors of the SOT and SLT dictionaries are ordered with respect to their energy level, which is found by calculating the variance of their coefficient values. The coefficients of block- and lapped-transform codecs are organized into 64-subbands given in Figure 11 depending on their energy levels [63, 76]. Basically, the numbers in the Figure 11 indicate the locations of the coefficients with respect to the energies of their basis vectors. In other words, while location 1 gets the highest energy coefficients of the blocks, the

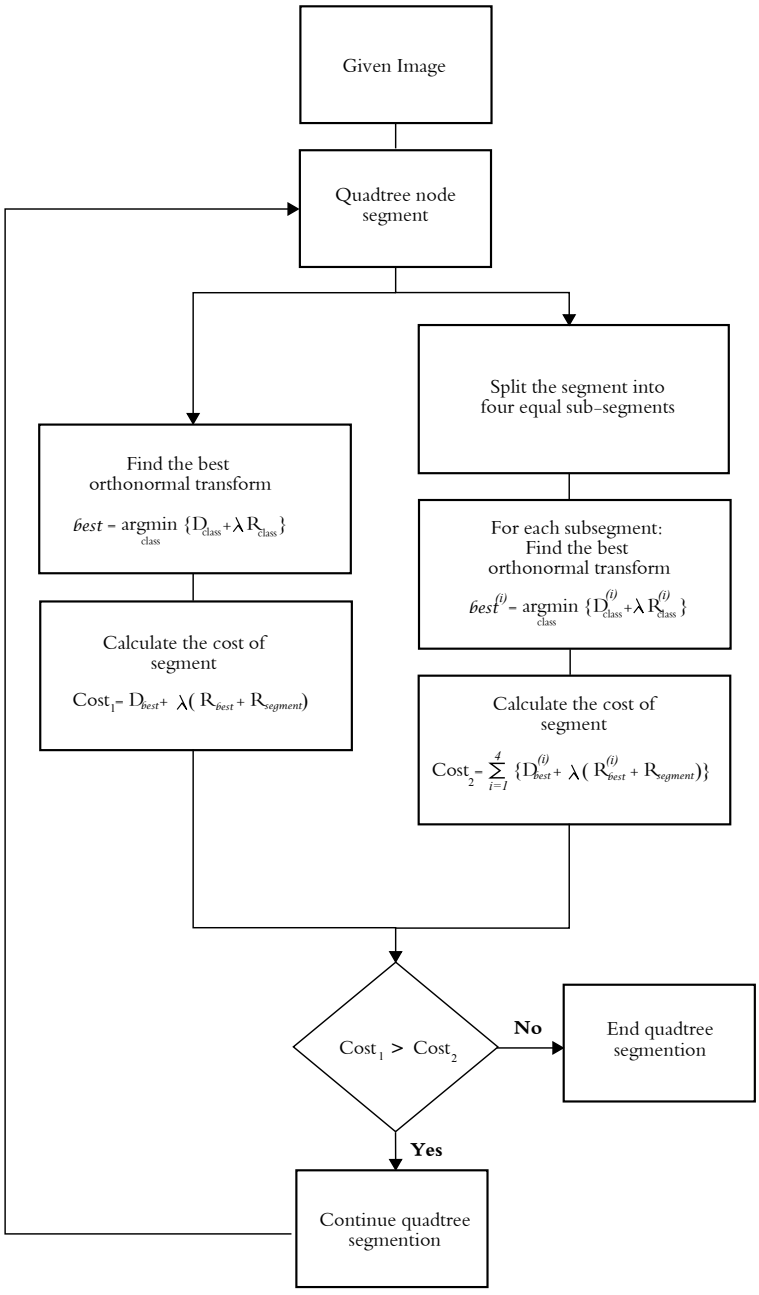


Figure 9: Flow-chart for quadtree segmentation.

location 64 gets the ones with lowest energy. A different approach is employed for wavelet-transform codec in which the coefficients with the same subband orientation are ordered in decreasing order of coefficient energies from coarse to fine scales in first 3-levels [61]. Since it is common to have 5-level wavelet decomposition, additional 2-level CDF 9/7 decomposition is applied to low frequency components of 3-level representation. Finally, the coefficients of the SOT, SLT and SMT are quantized with a uniform dead-zone quantizer which is followed by entropy coding with a SPIHT-like encoder.

The experiments presented in this paper are conducted with a standard set of test images. The size most of the images is 512×512 pixels except *foreman* and *cameraman* images, which are 256×256 pixels. Test images with names *vermeer, museum* and *chair* are computer generated and the rest is natural images. While *peppers, foreman, cameraman* and synthetic images have strong directional structure, *barbara* has a distinct anisotropic texture. *mandrill* has also complicated textural elements with some directionality. The compression results are given in Table 3. Note that the proposed methods (SOT,SLT, and SMT) always outperforms the conventional approaches (DCT,LBT, and CDF 9/7).

3.4.4 Adaptive Block Size

Another important aspect of the compression is adapting to the scale of the local structure. Wavelet transforms are quite successful in multiscale representation compared to block transforms. To get close to the compression performance of multiscale representation with the block transform, the support size of the transforms are adaptively changed depending on local structure. For example, around the fine image details the reduction of block size will help to capture local variations better. To implement this, SOTs with three different block sizes are trained (4×4, 8×8, and 16×16.) Next, these transforms are incorporated into block-based codec in which the

Table 1: Compression performances of block-, lapped- and wavelet-based codecs at 0.5 bits per pixel in terms of PSNR(dB).

	BLOCK		**LAPPED**		**WAVELET**	
	DCT	**SOT**	**LBT**	**SLT**	**D97**	**SMT**
lena	36.09	**36.45**	36.90	**37.02**	37.22	**37.39**
barbara	30.97	**31.26**	32.82	**32.85**	31.76	**32.24**
museum	36.29	**37.68**	37.02	**37.70**	37.95	**38.63**
mandrill	25.19	**25.41**	25.53	**25.54**	25.66	**25.86**
boat	32.42	**32.67**	33.07	**33.13**	33.24	**33.43**
vermeer	41.00	**41.76**	41.01	**41.70**	41.64	**42.32**
cameraman	30.65	**31.51**	30.73	**31.13**	31.50	**31.67**
foreman	37.11	**38.05**	37.99	**38.30**	38.67	**38.87**
chair	39.14	**40.64**	39.49	**40.24**	39.68	**40.64**
peppers	34.93	**35.21**	35.41	**35.60**	35.72	**35.84**
bridge	26.75	**26.84**	27.14	**27.20**	27.21	**27.35**
goldhill	32.54	**32.60**	33.11	**33.13**	33.14	**33.35**

quadtree segmentation is altered to accommodate block size adaptation. Basically, for each segment the block size of the transform that gives best sparsity-distortion cost is selected. Similar to Figure 11, the coefficients are ordered in depth-two-, depth-three-, and depth-four-tree formations for 4×4, 8×8 and 16×16 block sizes, respectively. Quantization and entropy coding is kept same as the fixed-block-size transform coder. The rate-distortion performance of DCT, SOT with fixed 8×8 block size, and SOT with block size adaptation is provided in Figure 13. Note that the block size adaptation gives significant RD gains, which are comparable with those of JPEG2000.

3.5 Conclusion

A block-based image compression algorithm capable of exploiting correlations along directional singularities is introduced. For this purpose, a transform design method that jointly optimizes the classification of blocks and corresponding transforms over a training set is presented. The result is a set of optimal transforms that replace

the traditional block transforms used in image compression. Although the geometry information is used only at the initialization of the transform optimizations, the resulting transforms still retain directional structure. Our compression results show significant gains compared to baseline systems on images having significant geometrical structure.

Another important contribution presented in this chapter is the design of directional lapped transforms, which does not require any knowledge about the complex modulation techniques. Sparse Lapped Transforms achieve gains up to 1dB PSNR improvement over the performance of Lapped Bi-orthogonal Transform, and it is one of the first directional lapped transform design in the literature.

This chapter also presents a sparsity-distortion-optimized multiresolution representation of image geometry. The proposed method uses the wavelet transform followed by a set of orthonormal transforms that are optimized for geometry. The designed orthonormal transforms locally adapt to the signal singularities in wavelet domain and provide a sparser representation. Rather than having a model-based approach, a data-driven training method is used to improve the performance of multiresolution wavelet representation. A new image codec is designed as an application of the proposed method which produces competitive image compression results with the state-of-the-art methods. Compared to similar foot-print and wedge-print methods, a consistent increase in rate-distortion performance is observed from low to high bitrates.

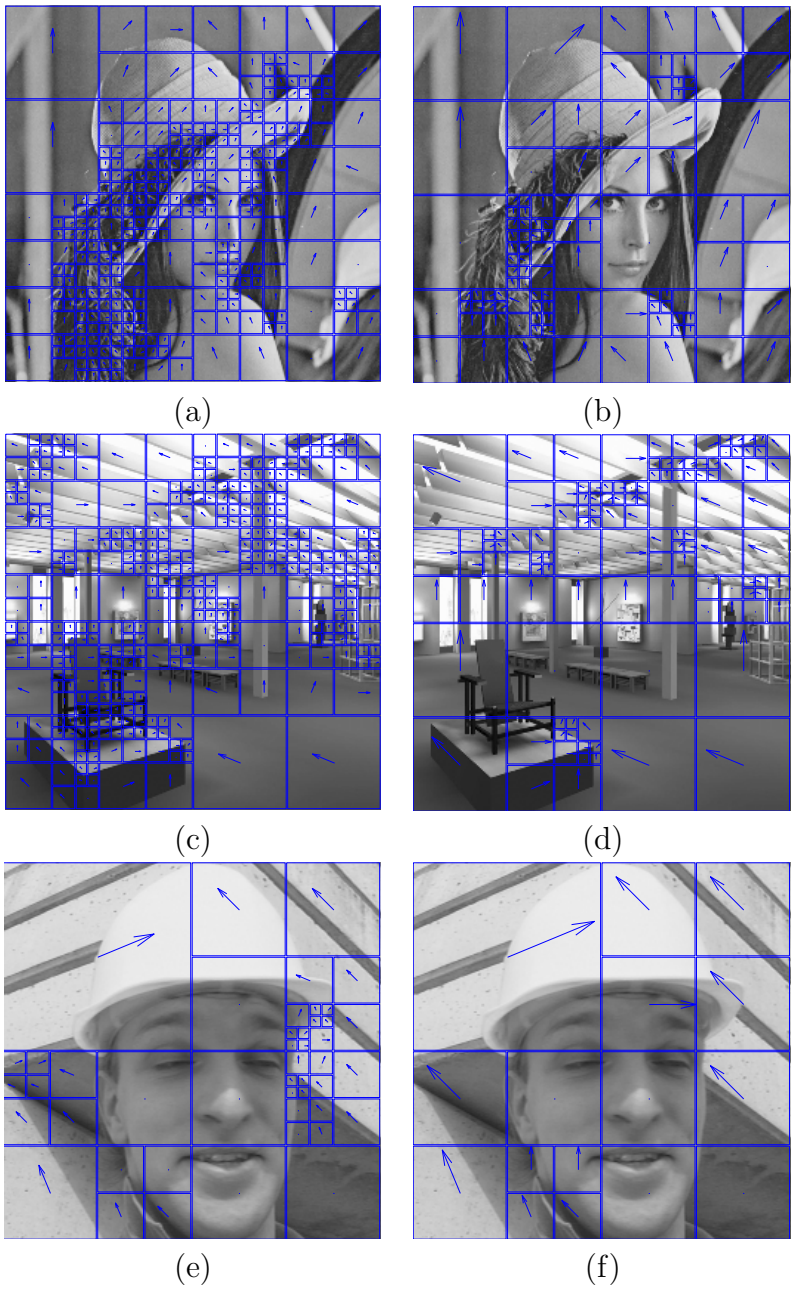


Figure 10: Quadtree classification results for $\lambda = 25^2$ (left column) and $\lambda = 50^2$ (right column) for images *lena* (top row), *museum* (middle row) and *foreman* (bottom row). Larger blocks indicate that all 8×8 blocks within utilize the same transform. The eight arrow directions correspond to the eight different optimized transforms and the dot symbol corresponds to the DCT.

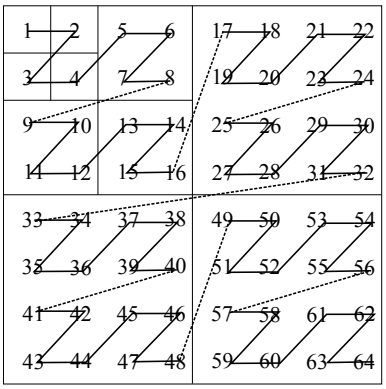


Figure 11: Order of 8×8 SOT and SLT coefficients in 64-subbands before entropy coding.

Figure 12: Standard test images used in the simulations. Top row; *lena*, *barbara*, *museum*, *mandrill*. Middle row; *boat, vermeer, cameraman, foreman*. Bottom row; *chair, peppers, bridge, goldhill*.

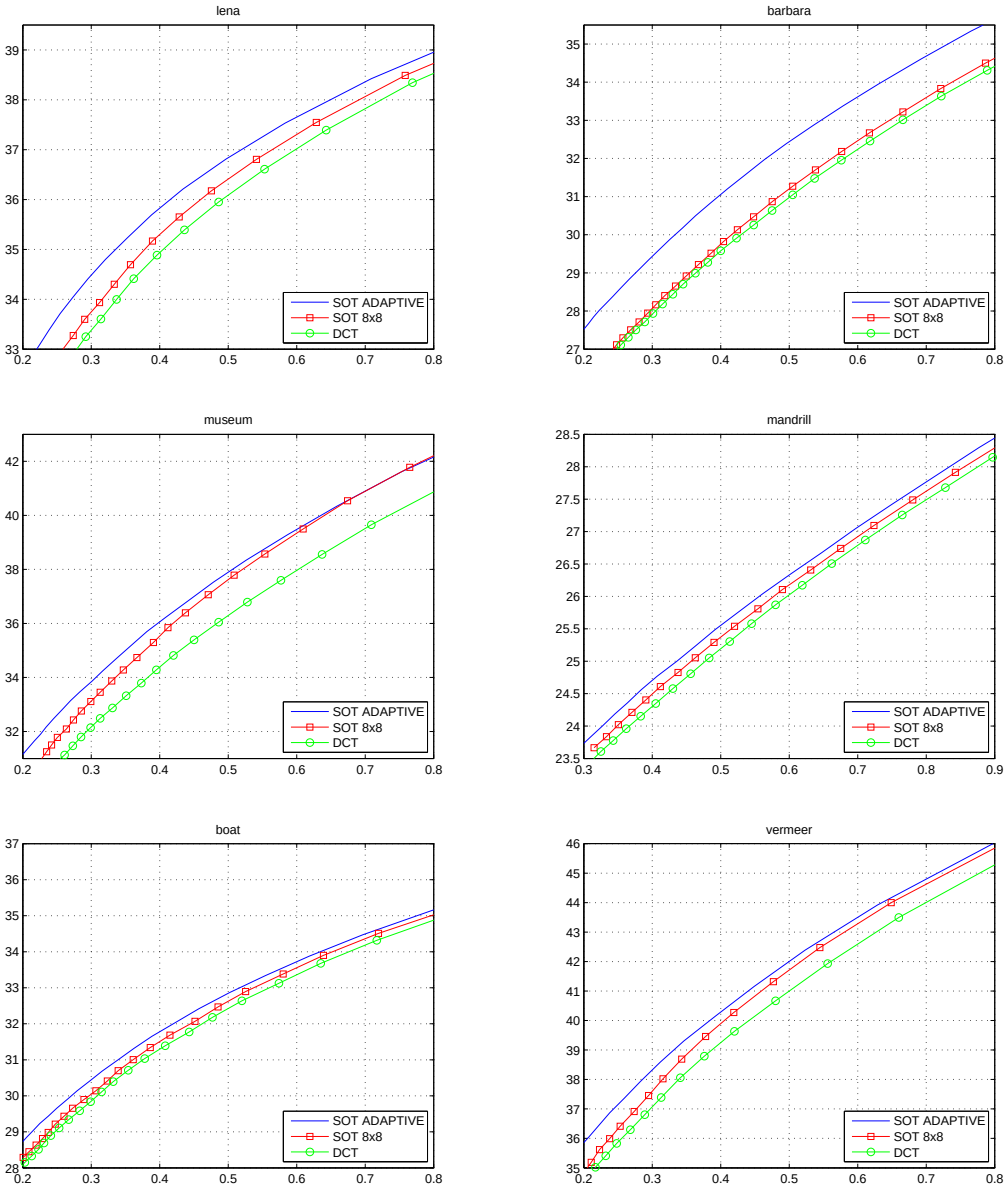


Figure 13: PART I- Rate-distortion curves for the test images in Figure 12.

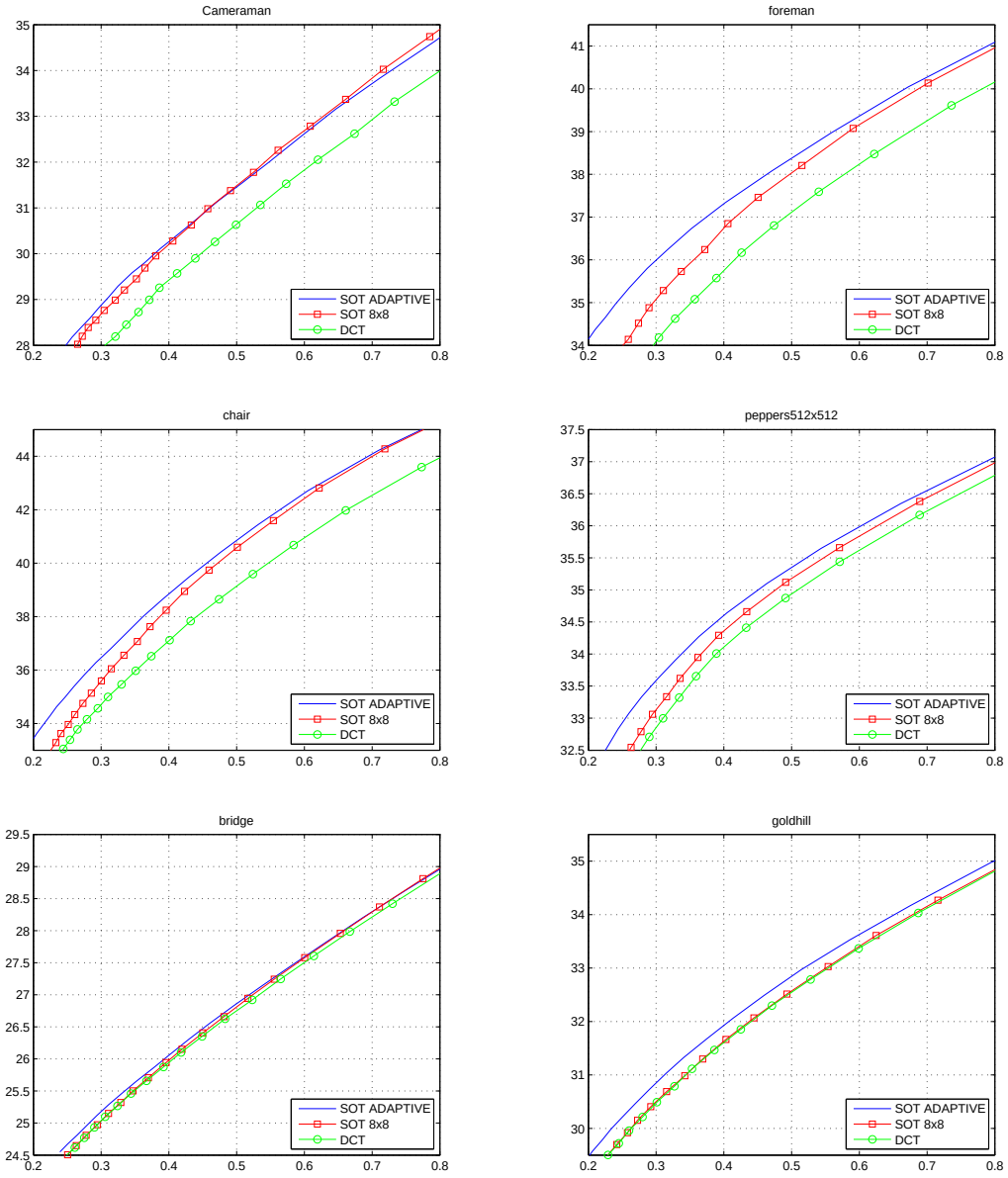


Figure 14: PART II- Rate-distortion curves for the test images in Figure 12.

CHAPTER IV

TRAINING-BASED 2-D NONLINEAR LIFTING

4.1 Introduction

Wavelets, as an image compression tool, play a significant role for better understanding of image data by providing multi-scale representations. Although traditionality wavelets are designed for 1-D signals, separable 2-D wavelet transforms are shown to achieve impressive compression performances [4] [60]. However, the separable transform approach used for the image data does not provide satisfactory results around the edges that are not aligned with vertical or horizontal directions. Recently, there have been many efforts to solve the shortcomings of the separable transform by utilizing the lifting scheme proposed by Sweldens in [69].

Briefly, the standard lifting architecture splits a 1-D signal into even and odd parts so that it can predict the odd samples from the even ones. Next, the residual of the prediction will be used to update even samples. This scheme is particularly useful for fast and in-place calculations of any discrete wavelet transforms (DWT) [20]. Lifting approach actually offers a different insight to wavelet analysis and provides more flexible means to design compression algorithms that perform better especially around the edges. Recently, very successful compression methods based on the lifting scheme have been proposed in [21] [14]. These methods essentially modify the prediction step in the original architecture into directional prediction such that the energy in the residual of the prediction (or detail coefficients) is minimized. Generally, the direction estimation is achieved either by block based rate-distortion (RD) optimization [45] [14] or by using a quad-tree partitioning [21]. The encoder then signals the directions to the decoder as side information.

In the literature, there are also lifting-based image compression methods that do not need to send any side information to the decoder side. Among the notable ones, Taubman [70] provided a non-separable approach for compression via lifting where the prediction is performed by adapting filters to the edge directions. Gerek *et al.* [33] developed a method that predicts a component by weighted sum of its polyphase neighbors where weights are updated by an adaptive LMS algorithm. This approach, however, suffers in low bitrates because it requires high-band coefficients to update the weights, which are prone to quantization. Claypoole *et al.*[15], on the other hand, proposed a nonlinear method which adapts the size of the wavelet filters in the prediction step such that the support of the filter will not cross the edge. This approach enabled them to have better prediction across the edges. Before that Donoho [23] presented the basic architecture of Claypoole *et al.*'s work in the context of average-interpolation which can be implemented within the lifting framework. This leads to the 1/N-tap branch of Cohen-Daubechies-Feauveau family (CDF) [16] biorthogonal wavelets with boxcar kernel as the dual function. Nevertheless, only subjective quality improvements are reported. These lifting-based methods do not require to indicate filter choices or direction information to the decoder. Therefore, the filters are adapted by utilizing the information in high bands [33] or in low bands [70] [15] which are the only available set of data at the decoder side.

Although the lifting method enable new ways to improve compression performance around the edges, 1-D polynomial interpolation scheme of wavelets has not been changed. Basically, the improvement in compression is achieved by finding the best direction to apply lifting that will reduce the energy in the high bands. Moreover, considering all the possible 2-D structures that one may observe in image data, 1-D directional lifting methods can only perform better around the strong edges while their performance in the textured regions stays limited. Therefore, we believe that lifting has not yet been fully utilized for these cases where better 2-D interpolation

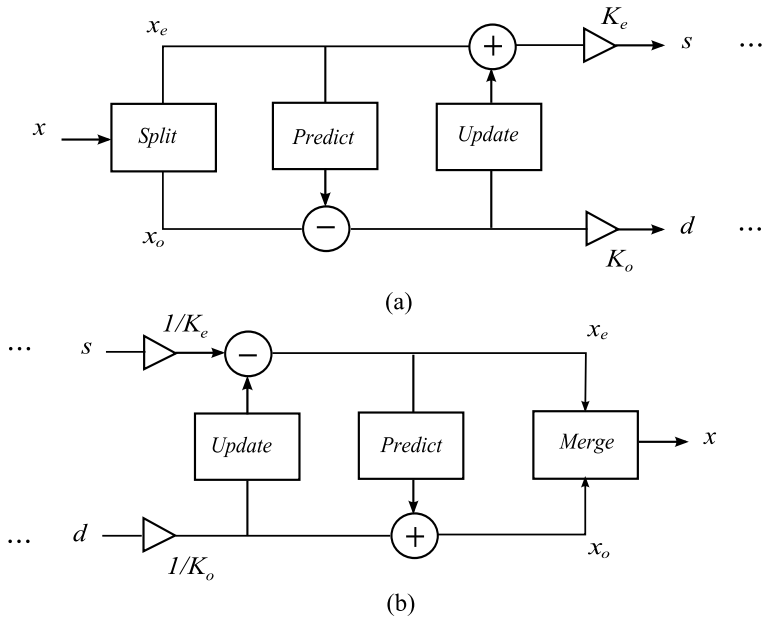


Figure 15: Single scale lifting scheme for forward (a) and backward (b) transforms.

schemes are needed.

In this chapter, we propose a new way towards generating interpolators for image compression problem that can take account the local 2-D content better for the lifting scheme. These new 2-D interpolators (or predictors) are obtained by a training method using Boxcar/Wavelet transform architecture [23]. Most of the filters that come out of the training process are observed to possess directional information together with some textural clues. Moreover, the prediction filters for the smooth regions become similar to the filters in the prediction step of 1/N-tap BWT. The proposed multi-scale image representation may help one to create better resolution enhancement methods as well.

4.2 2-D Wavelet Transform via Lifting

The first step of forward lifting transform is to split data into two polyphase components as show in Figure 15-(a). For the input signal x, these two components x_o and x_e are named as odd and even subsets of x, respectively. For 2-D case, we have $x_e[m,n] = x[2m,n]$ for even and $x_o[m,n] = x[2m+1,n]$ for odd subset of the image

rows. Next, the odd samples are predicted from the even ones and high-pass subband samples are calculated as follows:

$$d[m,n] = x_o[m,n] - \sum_i h_i x_e[m+i,n] \tag{33}$$

where h_i's corresponds to the prediction filter coefficients. For the update step to extract low-pass subband samples we have

$$s[m,n] = x_e[m,n] + \sum_j l_j d[m+j,n] \tag{34}$$

where l_j's are update filter coefficients. The details to calculate h_i and l_j for different wavelet transforms are given in [20]. Forward transform ends with a final normalization step for keeping energies of the high and the low bands same in different scale. Backward transform can be achieved by performing the opposite operations in the reverse order.

One way to improve the compression performance of the lifting algorithm is to use nonlinear predictors instead of linear ones. However, there is no clear way to update the other polyphase component after this prediction. From this, Claypoole *et al.* [15] suggested to reverse the order of prediction-update of standard lifting into update-predict lifting. This architecture is discussed in [23] as Boxcar/Wavelet transform in the context of multi-resolution average-interpolation. In this chapter, the same update-predict lifting scheme is used to improve compression performance of standard wavelets via nonlinear prediction.

4.3 Adaptive Boxcar/Wavelet Transform

The proposed algorithm has two designing aspects. The first one is to create the compression model based on Boxcar/Wavelet Transform. Later, the difference between Boxcar/Wavelet and our method will be clear. The second aspect is to generate a set of context dependent filters that are optimized for nonlinear prediction.

4.3.1 Image Compression Model

Starting from our compression model; consider an image as a 2-D continuous function $f(x, y)$. Next, the following discretization of the continuous data is common for image processing,

$$s^j_{m,n} = \int\int \chi^j_{m,n}(x, y) f(x, y) dxdy \equiv < f, \chi^j_{m,n} > \tag{35}$$

where

$$\chi^j_{m,n}(x, y) = 2^{2j} \mathbf{1}_{[m/2^j,(m+1)/2^j]\times[n/2^j,(n+1)/2^j]} \tag{36}$$

is called boxcar function (or kernel). From image processing point of view, j can be thought as the resolution of the image data, m and n are the row and column indices of the observed pixel, respectively. Therefore the following can be written for the relations between the resolutions (or scales),

$$s^{j-1}_{m,n} = \frac{1}{4}(s^j_{2m,2n} + s^j_{2m+1,2n} + s^j_{2m,2n+1} + s^j_{2m+1,2n+1}). \tag{37}$$

Note that as j decreases, the resolution gets coarser. Now consider the discretization given in Equation (35) be the low-pass samples for the compression scheme. Then one can write a 2-D separable transform similar to Boxcar/Wavelet transform defined in [23]. The coefficients of wavelet transform is found as follows:

$$s^{j-1}_{m,n} = \frac{1}{2}(s^j_{2m,n} + s^j_{2m+1,n}) \tag{38}$$

$$d^{j-1}_{m,n} = s^j_{2m+1,n} - (s^{j-1}_{m,n} + (s^{j-1}_{m+1,n} - s^{j-1}_{m-1,n})/8). \tag{39}$$

Here $s^{j-1}_{m,n}$ and $d^{j-1}_{m,n}$ are low-pass and high-pass coefficients at scale $j-1$, respectively, which are obtained by the column transform. Extraction of low-pass averages given in Equation (38)corresponds to the *update* step in the lifting algorithm. Likewise, the high-pass coefficients in Equation (39) are the residuals of the *prediction* of actual low-pass coefficients from its neighbors. The details of different prediction kernels can be found in [23] and [16]. We have used 1/3-tap BWT just for illustration purposes.

Finally, the low-pass and high-pass coefficients are normalized to preserve the energy between the scales.

The proposed approach replaces the one dimensional prediction, which uses polynomial of order D (note $D = 2$ for Equation (39) with a two dimensional prediction. Thus, Equation (39) will become,

$$d_{m,n}^{j-1} = s_{2m+1,n}^{j} - \sum_{k,l \epsilon N} H_{\{k+D/2,l+D/2\}}^{j-1} s_{m+k,n+l}^{j-1} \tag{40}$$

where for even D, $N = \{-D/2, .., 0, .., +D/2\}$ and H^{j-1} is the new two dimensional prediction filter at scale $j-1$. In order to find the optimum 2-D prediction, a training-based algorithm is designed. Moreover, rather than a fixed prediction filter, depending on the feature vector extracted from low-pass coefficients, the new 2-D filters are adapted to the local content similar to the resolution synthesis interpolation technique that is proposed in [5].

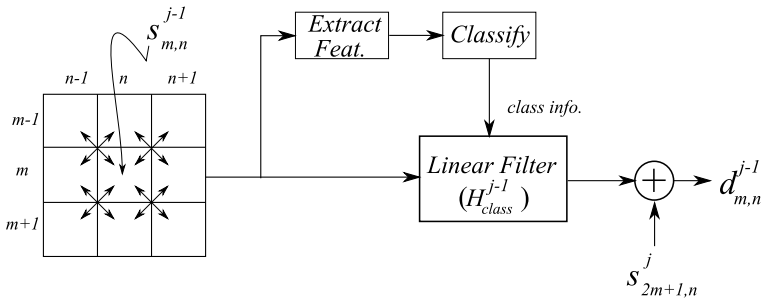


Figure 16: Predict step for column transform.

4.3.2 Optimum Filter Design

To adapt filters in Equation (40), we proposed a feature-based context classification method. The proposed method finds the *optimum* filters for various contents of low-pass coefficients via an off-line training algorithm.

In the training phase features are extracted from low-pass coefficients of the given scale, let's say scale $j-1$. The following feature vector is proposed to identify the

context around these low-pass coefficients at scale $j-1$, $s_{m,n}^{j-1}$:

$$f(\mathcal{N}_{s_{m,n}^{j-1}}) = [|s_{m-1,n-1}^{j-1} - s_{m,n}^{j-1}|, |s_{m,n-1}^{j-1} - s_{m-1,n}^{j-1}|, \ldots, \\ |s_{m,n}^{j-1} - s_{m+1,n+1}^{j-1}|, |s_{m+1,n}^{j-1} - s_{m,n+1}^{j-1}|]/\mathcal{K} \tag{41}$$

where $\mathcal{K}$ is a normalization factor and $\mathcal{N}_{s_{m,n}^{j-1}}$ denotes 3×3 neighborhood of $s_{m,n}^{j-1}$ that is shown on the left at Figure 16. Also note that the entries of feature vector, f, correspond to the absolute differences between the coefficients shown with double-headed arrows in the Figure 16.

The feature vectors extracted from the training data set are grouped into M different context classes by a *K-means* clustering algorithm. Next, the optimum filters for these classes are found by solving the following minimization,

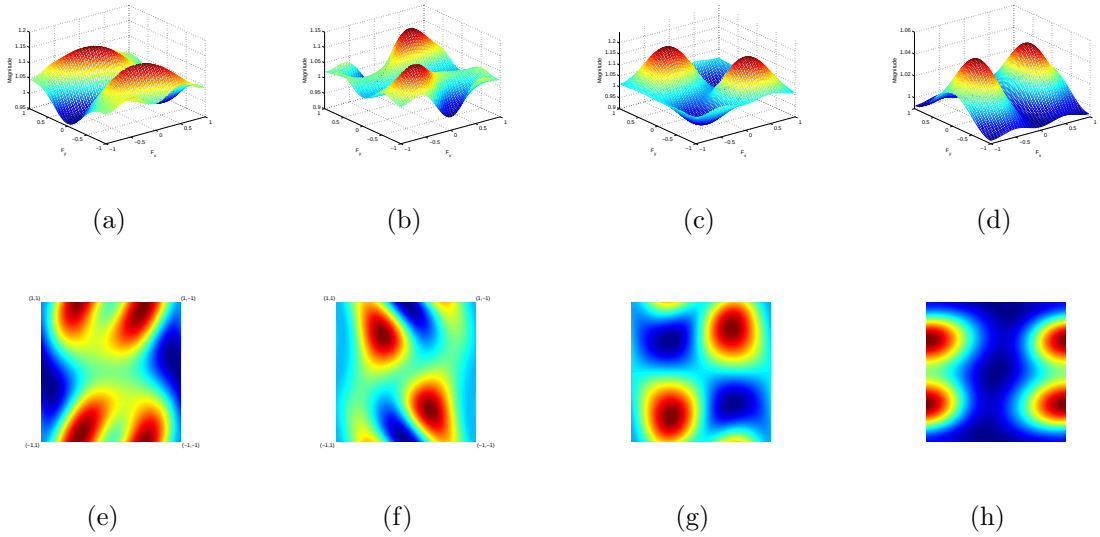
$$H_{Class}^{j-1} = \arg\min_{G} \sum_{(m,n)\epsilon\mathcal{C}} [s_{2m+1,n}^{j} - \sum_{k,l\epsilon N} G_{\{k+D/2,l+D/2\}} s_{m+k,n+l}^{j-1}] \tag{42}$$

where $\mathcal{C} = \{(m,n)|f(\mathcal{N}_{s_{m,n}^{j-1}}) \; \epsilon \; Class\}$ and G is 2-D filter function. Note that, there will be M filters for M context class. Next, these new 2-D prediction filters are adaptively used in Equation (40). The adaptation is determined by the context of the neighborhood of a low-pass coefficient of each scale as shown in Figure 16. This procedure is expected to reduce the energy in the high-pass coefficients $d_{m,n}$, which is desirable to have improved image compression performance.

4.4 Results and Discussions

Before testing the performance of the proposed method, first a set of natural images are selected for training. 20 context classes are defined and corresponding 2-D prediction filters of size 3×3 are obtained by the method discussed in Section 4.3.2. Figure 17 shows frequency response of sample filters. Note that the filters possess directional information together with some textural clues. Since the feature vectors are extracted from the low-pass coefficients of each scale, the effect of quantization to these coefficients has to be considered to have the same adaptation in the decoder

. To achieve this, the effect of quantization is precalculated in the encoder side such that the decoder can recover the same filter adaptation that is used in the encoder. Next step is to compress the test images with three scale row and column transforms, which is then followed by the quantization of the coefficients by a uniform dead-zone quantizer. Finally, the quantization bins are entropy coded with SPIHT [60].



(a) (b) (c) (d)

(e) (f) (g) (h)

Figure 17: Frequency response of prediction filters obtained by training. Note figures in the bottom row corresponds to top-views of the figures in the top row.

In Figure 18, the results of the proposed compression algorithm together with 9/7-tap BWT is given. Observe that the feature-set given in Equation (41) favors directional edges. Also the subjective quality improvement around the eye and hat in Figure 18-(b) compared to Figure 18-(a) is noticeable. Since the support of our filter is smaller, ringing artifacts on the strong edges are reduced in Figure 18-(d) compared to 18-(c).

Our experiments indicate that the proposed method outperforms 5/3-tap BWT both in subjective and objective quality. Refer to Figure 19 for rate-distortion comparisons between different methods for cameraman image. Since the proposed architecture is similar to 1/3-tap BWT, we added its RD curve as well.

(a)

(b)

(c)

(d)

Figure 18: Lena detail compressed with 9/7-tap BWT (a), and with the proposed method (b) at 0.125 bpp. Cameraman image compressed with 9/7-tap BWT (c), and with the proposed method (b) at 0.5 bpp.

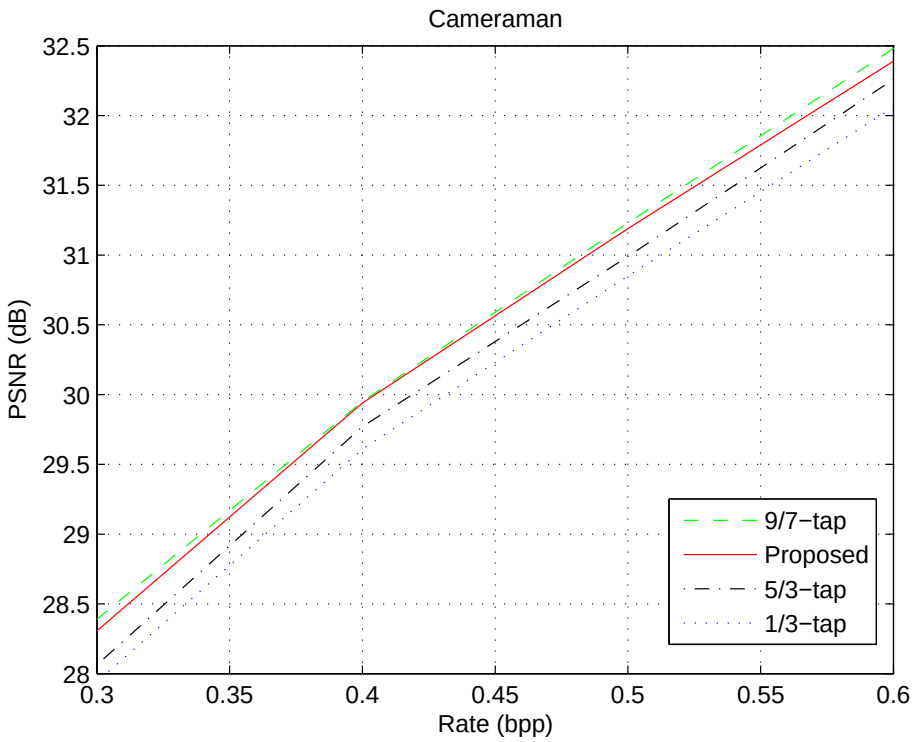


Figure 19: RD curve for cameraman image.

4.5 Conclusion

This chapter presents a new adaptive Boxcar/Wavelet transform. The proposed method differentiates itself from the ones in the literature by using lifting architecture to produce training-based two dimensional prediction filters. This approach not only provides fewer wavelet coefficients around the edges but also reduces the overall variations in the high-bands. We observe subjective quality improvement over standard bi-orthogonal transforms together with comparable objective quality. Moreover, the proposed multi-scale representation can be extended to generate new resolution enhancement methods as well.

CHAPTER V

MODE-DEPENDENT SPARSE TRANSFORMS FOR VIDEO CODING

An integral part of this research is to propose transforms for next generation video coding. Initial experiments show promising results and provide directions for new research topics mainly in two directions. First one is to reduce implementation complexity of transform evaluation process by means of a separable filter design. Essentially, the success of DCT-based video coding can be attributed to its decorrelating power and the simplicity of its evaluation with separable transforms. However, the block transforms that are presented up to this point (i.e., SOT) are non-separable with requires larger memory size to keep the transform coefficients. This observation is an important problem that needs to be addressed for DSP on-chip implementations, which mainly use line buffers for arithmetic operations and has limited memory.

One drawback of the separable application of DCT for video coding stems from not differentiating the columns of a block from its rows. As a result, only vertical and horizontal directions and smooth regions are represented efficiently. However, to improve the coding performance of the current block-based image and video codecs, the transforms should posses some anisotropic features that can adapt to the local characteristics of the image by using different column and row transforms. This objective can be achieved by designing separable transforms that are designed to be directional. The extension of SOT to separable classes of directional transforms can

provide this aspect by solving the following formula,

$$\min_{\mathbf{V}_k,\mathbf{H}_k} \left\{ \sum_{\mathbf{x}^j \in \mathcal{S}_k} \min_{\alpha_k^j} \|\mathbf{x}^j - \mathbf{V}_k \alpha_k^j \mathbf{H}_k^T\|_2^2 + \lambda \|\alpha_k^j\|_0 \right\} \tag{43}$$
$$s.t\ \mathbf{V}_k^T \mathbf{V}_k = \mathbf{I}, \quad \mathbf{H}_k^T \mathbf{H}_k = \mathbf{I}$$

where $\mathbf{H}_k$ and $\mathbf{V}_k$ are orthonormal transforms for rows and columns of block $\mathbf{x}^j$, respectively. The coefficients are found by $\alpha_k^j = \mathbf{V}_k^T \mathbf{x}^j \mathbf{H}_k$.

5.1 Introduction

This chapter describes a novel mode-dependent design of 2-D separable transforms to be used in video coding. Compared to the current state of the art methods, this method enforces sparsity on the transform coefficients for given data fidelity. Iterative optimization updates each separable transform after finding the optimal coefficients in a mode-dependent framework. The mode-dependent aspect of the transform design also deviates from prior work described in Chapter 3, in which the rate-distortion-optimal selection of transforms is abandoned. Hence, no extra bits are required to signal the transform selection, which makes our approach compatible with current video coding architectures.

In video coding, frames are typically encoded in two ways: i) intra coding, ii) inter coding. In intra coding the correlation of blocks within a frame is utilized to generate prediction residuals, which will have significantly less energy than the corresponding original image blocks. The prediction residual is the difference between an original block and its prediction. Hence, fewer bits are required to encode the blocks at a given level of fidelity. For inter coding, motion-compensated prediction residuals are generated using blocks within a temporal neighborhood.

In state-of-the-art video codecs such as H.264/AVC, the prediction for an intra

coded block is computed from previously coded neighboring blocks. Several directional predictions are generated, and a fitness measure such as sum of absolute differences (SAD), sum of squared error (SSE), or sum of absolute transformed differences (SATD) is computed for each direction. In H.264/AVC, the best prediction direction or "mode" is selected, and the corresponding prediction residual is transformed via the conventional integer Discrete Cosine Transform (DCT) prior to quantization. Since the residuals of the same mode possess common patterns of correlation, one can design transforms that will further exploit these patterns to reduce the bitrate. One such set of transforms are the Mode-Dependent Directional Transforms (MDDT) proposed in [78]. While MDDT utilizes the KLT or Singular Value Decomposition (SVD) to learn 2-D separable transforms for residuals of each intra prediction mode, this chapter describes shortcomings of KLT in the presence of outliers in the training data. Next, a new $\mathcal{L}_0$-norm regularized optimization method is proposed as a more robust way to learn 2-D separable transforms for video coding. By employing new transforms, which are termed as Mode-Dependent Sparse Transforms (MDST), into H.264/AVC-based video codec (JM11.0KTA2.6r1), the compression efficiency is improved by up to 3.9% BD-rate compared to MDDT, while the coding architecture is kept the same.

The outline of the chapter is as follows. In the next section, we point out why KLT is vulnerable to outliers in the data, and show how $\mathcal{L}_0$-norm regularization can bring robustness to the transform learning process. Section 5.3 outlines the proposed iterative optimization method used to generate 2-D separable transforms for video coding, which is followed by Section 5.4 where we introduce a new ordering method for the locations of the coefficients to improve coding efficiency of the entropy coder. In Section 5.5, experiments to validate the proposed method are provided. Finally, we make some concluding remarks in Section 5.6.

5.2 Learning Transforms from Data

Given a set of random signals, KLT is the standard procedure to extract transforms that will decorrelate the data to a smaller number of variables. With the KLT, the signal energy is concentrated mostly to the first few coefficients of this linear orthogonal decomposition, such that a reduced dimensional representation is achieved within certain fidelity. KLT solves the following minimization to find the principal component $\mathbf{g}_1$

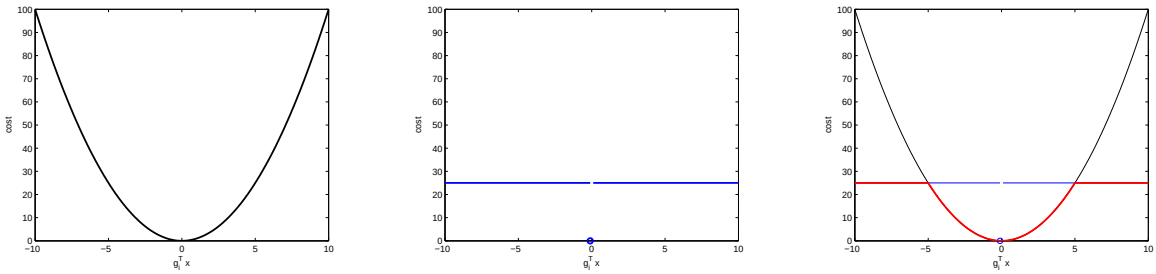
$$\min_{\Phi_1} \sum_{j \in \mathcal{S}} \|\mathbf{x}^j - \Phi_1 c_1^j\|_2^2 \;\; s.t \;\; \Phi_1^T \Phi_1 = 1, \tag{44}$$

where $\mathbf{x}^j$ is the j-th vector of size $n \times 1$ in the dataset $\mathcal{S}$, and c_1^j is the coefficient of the principal component. The principal vector aligns itself to the direction of maximum variation, and the solution can be found by using singular value decomposition (SVD). Similarly, the subsequent k-th components can be found from the residual data after the subtraction of the first $k-1$ principal components. Another way to express the KLT formulation is as follows:

$$\min_{\Phi} \sum_{j \in \mathcal{S}} \|\mathbf{x}^j - \Phi \alpha^j\|_2^2 \;\; s.t \;\; \Phi^T \Phi = \mathbf{I}, \tag{45}$$

where Φ_1 is the first column of matrix Φ. One of the problems with KLT-based learning arises from its noise intolerance. The least square norm in (44) is prone to outliers, especially to the ones with large energy. These outliers can arbitrarily skew the direction of the principal component. In cascade, the subsequent components and the overall performance of this representation will be affected. In computer vision and statistics literature there are several methods proposed to overcome this challenge, such as outlier rejection [77], weighted least squares [65], and utilizing robust error norms to learn subspaces [42].

In compression, recent learning-based designs have been shown to provide superior performance compared to standard methods such as the DCT or wavelets. Ye and



(a) (b) (c)

Figure 20: Cost functions of (a) $\mathcal{L}_2$ norm, (b) $\mathcal{L}_0$ norm, and (c) $\rho(.)$ as a function of $\Phi_i^T \mathbf{x}$ for fixed $\lambda = 25$ in (48).

Karczewicz [78] proposed to use KLT to learn 2-D separable transforms for video coding. In Chapter 3, a sparsity enforced transform designs, called Sparse Orthonormal Transforms (SOT), is presented. Apart from the iterative update of the clusters and the corresponding transforms, the Sparse Orthonormal Transforms (SOT) provides a learning algorithm that is more robust than the KLT, by regularizing the cost in (44) with the sparsity of the coefficients [62].

To be more specific, let Φ be of size $N \times N$. A robust estimation of the principal components can be achieved when the following cost is minimized

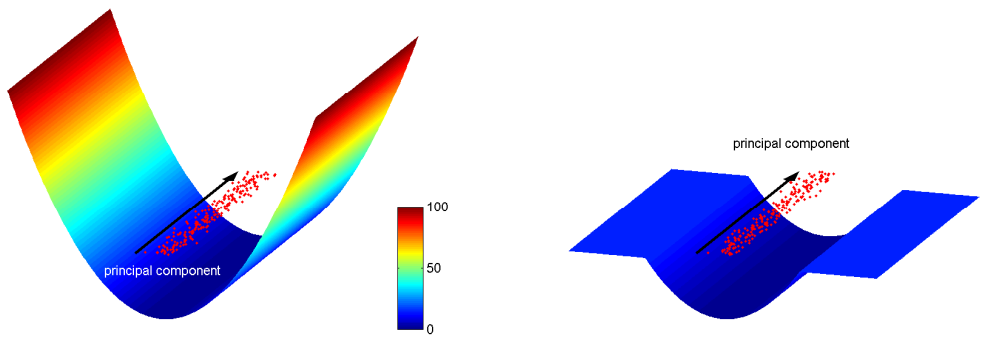
$$\min_{\Phi} \sum_{j \in \mathcal{S}} \min_{\alpha^j} \{ \|\mathbf{x}^j - \Phi \alpha^j\|_2^2 + \lambda \|\alpha^j\|_0 \} \;\; s.t \;\; \Phi^T \Phi = \mathbf{I}, \tag{46}$$

where α^j is the coefficient of Φ for data vector $\mathbf{x}^j$, λ is Lagrange multiplier, and $\|.\|_0$ is the $\mathcal{L}_0$ norm, which is equivalent to the the number of nonzero elements. Next, (46) can be written as

$$\min_{\Phi} \sum_{j \in \mathcal{S}} \sum_{i} \min_{\alpha_i^j} \{ (\Phi_i^T \mathbf{x}^j - \alpha_i^j)^2 + \lambda \|\alpha_i^j\|_0 \} \;\; s.t \;\; \Phi^T \Phi = \mathbf{I}, \tag{47}$$

where Φ_i is the i-th column of Φ, and α_i^j denotes i-th coefficient of vector $\mathbf{x}^j$. The cost defined in (47) penalizes nonzero α_i's; thus enforcing a sparse representation for component Φ_i. Note that first minimization term can be expressed as a function as follows:

$$\rho(\Phi_i^T \mathbf{x}^j, \lambda) = \min_{\alpha_i^j} \{ (\Phi_i^T \mathbf{x}^j - \alpha_i^j)^2 + \lambda \|\alpha_i^j\|_0 \}. \tag{48}$$



(a) (b)

Figure 21: Cost function of KLT (a), $\mathcal{L}_0$-norm regularized solution (b), and their corresponding principal components.

where ρ is a cost function of two variables. Essentially, ρ is a union of $\mathcal{L}_2$ and $\mathcal{L}_0$ norms. For small values of α_i, the $\mathcal{L}_2$ norm is active, whereas the $\mathcal{L}_0$ norm dominates the function for larger values of α_i. The transition between two norms is defined by λ. Figures 20(a) and 20(b) show the $\mathcal{L}_2$ and $\mathcal{L}_0$ norms as a function of $\Phi_i^T\mathbf{x}$. Figure 20(c) plots $\rho(\Phi_i^T\mathbf{x}, \lambda)$, which picks the minimum of these norms for given $\Phi_i^T\mathbf{x}$ and λ values.

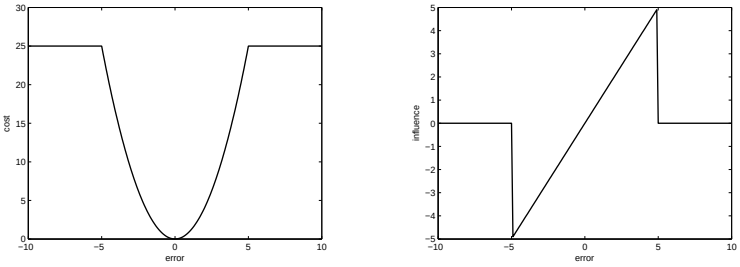
The minimization over the coefficients can be substituted by the M-estimator $\rho(.)$ as follows:

$$\min_{\Phi} \sum_{j \in \mathcal{S}} \sum_{i} \rho(\Phi_i^T \mathbf{x}^j, \lambda) \quad s.t \;\; \Phi^T\Phi = \mathbf{I}. \tag{49}$$

Due to orthonormality conditions imposed on the solution, this expression essentially searches for the axis rotations that will minimize the cost function $\rho(.)$ over a set of observations. Sparsity imposed on a component helps robust estimation of components orthogonal to that. To visualize this, Figure 21 gives a $3D$ perspective of the $\mathcal{L}_2$ and $\mathcal{L}_0$-regularized cost functions used in (44) and (49) for 2-D data. If Φ_1 assumed to be the principal component, the cost function in Figure 21(b) is attained by imposing sparsity on the coefficients of Φ_2, where $\Phi_1 \perp \Phi_2$. Here we show how the principal components should be aligned with respect to given data (dots in 2-D) to minimize the costs. Note that even a single large outlier would arbitrarily change

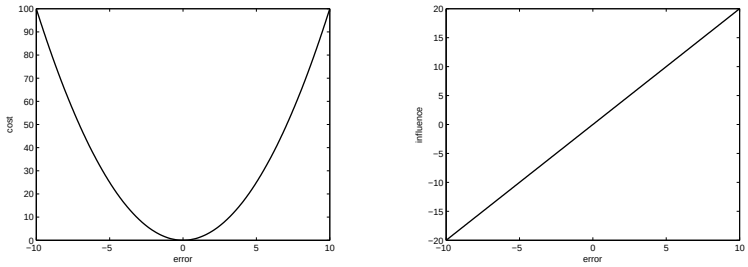
the direction of KLT-solution shown in Figure 21, due to rapid increase of the cost function.

To better understand the effects of outliers to the error norm minimization problem, the influence functions of various error norms used in robust statistics literature are plotted in Figures 22 to 24. Note that the error norm of KLT is given in Figure 23, where the corresponding derivative function shows that the influence of an outlier gets linearly larger. On the other hand, the outliers have no influence when the error function is regularized with $\mathcal{L}_0$-norm, which is given in Figure 22. The robustness of $\mathcal{L}_0$-norm regularized SOT solution comes from this reality. Also in order to avoid local minima, annealing λ is a common approach in robust statistics literature [9, 10]. A linear regression experiment with outliers is provided in Section 5.5 to compare the robustness of standard KLT and $\mathcal{L}_0$-norm regularized solution.



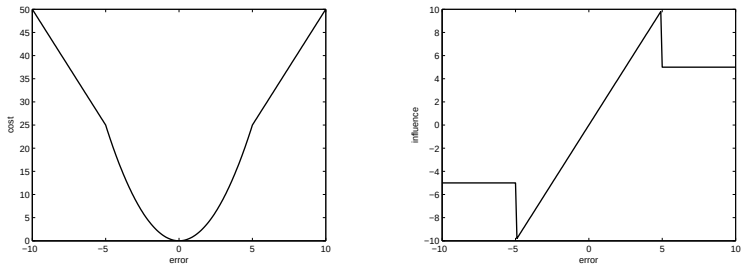
(a) (b)

Figure 22: The cost function, $\rho(\text{error}, 25)$, for L0 norm (a), and its derivative (or influence) (b).



(a) (b)

Figure 23: The cost function, $\rho(\text{error})$, for L2 norm (a), and its derivative (or influence) (b).



(a) (b)

Figure 24: The cost function, $\rho(\text{error}, 25)$, for L1 norm (a), and its derivative (or influence) (b).

5.3 Mode-Dependent Sparse Transforms (MDST)

There are two standard approaches for block-based 2-D data transforms: i) separable, and ii) non-separable transforms. In the separable case, each column and row of the block is considered as a 1-D signal, and 1-D transforms are used to map the block of data to a set of coefficients. The 1-D transforms used in each direction could be the same, but may also be different. For non-separable transforms, the block is generally ordered as a 1-D vector by lexicographically ordering columns or rows of the block. The disadvantage of this is that non-separable transforms would require more memory to hold the entries of the transform matrix. Also, large matrix multiplications are generally too complex for hardware implementations. Therefore, separable transforms are appealing. However, there is a cost for separable transforms, since they only utilize the correlation with a column or row; hence the compression performance of the separable transforms is lower around directional edges as compared to non-separable transforms.

Intra coding of H.264/AVC has been shown to provide higher coding efficiency compared to standard block based image compression methods such as JPEG, and it has competitive performance with, if not better than, wavelet based JPEG2000 [66, 54]. The success is largely due to intra prediction methods employed prior to transform coding. In general, the residual data generated by intra prediction has less

energy than the original image block, hence requires fewer bits to represent coefficients after transform coding. Nevertheless, even after the intra prediction, residuals are observed to possess directional structures often aligned with the direction of prediction. Therefore for each directional prediction mode a new transform is trained in [78] to further utilize the inherent structure of that prediction mode to reduce the bitrate. We will improve upon that transform design process with a new iterative optimization method to learn 2-D separable transforms for each intra prediction mode.

We define the number of prediction modes as M, where $M = 9$ for intra prediction of 4×4 and 8×8 block sizes, and $M = 4$ for 16×16 blocks. For each mode, two separable transforms are needed. The vertical and horizontal transform for mode i is denoted as $\mathbf{V}_i$ and $\mathbf{H}_i$, respectively. Let the $N \times N$ block $\mathbf{X}_i^j$ be the j-th residual block encoded using intra mode i, and α_i^j be the corresponding coefficient matrix of the residual signal. The sparsity-distortion cost function can be written as follows:

$$
\begin{aligned}
& i \in \{1, ..., M\} : \\
& \min_{\mathbf{V}_i, \mathbf{H}_i} \left(\sum_{j \in S_i} \min_{\alpha_i^j} \|\mathbf{X}_i^j - \mathbf{V}_i \alpha_i^j {\mathbf{H}_i}^T\|_2^2 + \lambda \|\alpha_i^j\|_0 \right) \\
& s.t \ {\mathbf{V}_i}^T \mathbf{V}_i = \mathbf{I}, \ \ {\mathbf{H}_i}^T \mathbf{H}_i = \mathbf{I} \, .
\end{aligned} \tag{50}
$$

To learn the transforms for mode i, we have formed a training dataset S_i, over which the cost function will be minimized. The given cost models distortion as the reconstruction error (first term in the summation), and an approximation to rate is given by $\mathcal{L}_0$ norm term, which is the number of nonzero coefficients. In Section 5.2 we have also pointed out how $\mathcal{L}_0$-norm regularization relates to robust estimation. The proposed method iteratively finds optimal coefficients and updates one of the separable transform at each iteration. Let us assume vertical and horizontal transforms are initialized with the DCT, then for the i-th mode we apply the following steps:

I. *Optimal coefficients for a given transform:* The sparsest representation for a given transform can be found by solving

$$\alpha_i^{j^*} = \arg\min_{\mathbf{D}} (\|\mathbf{X}_i^j - \mathbf{V}_i \mathbf{D}_i^j {\mathbf{H}_i}^T\|_2^2 + \lambda \|\mathbf{D}_i^j\|_0). \tag{51}$$

Note that since both $\mathbf{V}_i$ and $\mathbf{H}_i$ are orthonormal, the optimal solution is obtained by hard-thresholding the components of $\mathbf{D} = {\mathbf{V}_i}^T \mathbf{X}_i^j \mathbf{H}_i$ with $\sqrt{\lambda}$.

II. *Optimal vertical transforms for given coefficients:* The optimal vertical separable orthonormal transform for given coefficient vectors from previous step can be found by solving

$$\mathbf{V}_i^* = \arg\min_{\mathbf{A}} \left\{ \sum_{\mathbf{X}_i^j \in S_i} \|\mathbf{X}_i^j - \mathbf{A} {\alpha_i^j}^* {\mathbf{H}_i}^T\|_2^2 \right\} \tag{52}$$

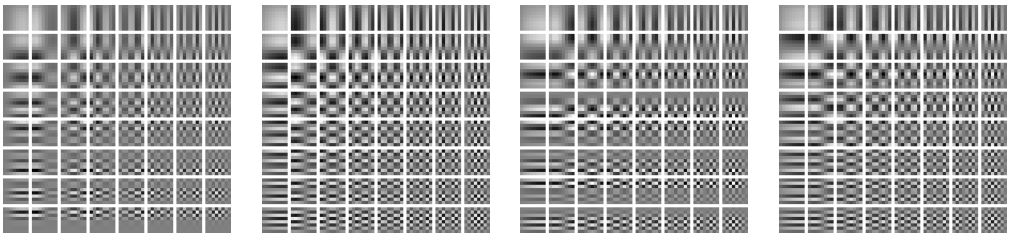
$$s.t. \quad \mathbf{A}^T \mathbf{A} = \mathbf{I}.$$

Note here that the horizontal separable transform, $\mathbf{H}_i$, is assumed to be fixed. Let $\mathbf{Y} = \sum_{\mathbf{X}_i^j \in S_i} {\mathbf{X}_i^j}^T {\mathbf{H}_i}^T \alpha_i^j$, and its SVD be $\mathbf{Y} = \mathbf{U} \Lambda^{1/2} \mathbf{W}^T$. The solution for the optimal orthonormal transform can be found by $\mathbf{V}_i^* = \mathbf{W}\mathbf{U}^T$. For details of the optimization please refer to [63].

III. *Optimal coefficients with updated vertical transform:* This time optimal coefficients are found with optimized transform, $\mathbf{V}_i^*$, from the previous step,

$$\alpha_i^{j^*} = \arg\min_{\mathbf{D}} (\|\mathbf{X}_i^j - \mathbf{V}_i^* \mathbf{D}_i^j {\mathbf{H}_i}^T\|_2^2 + \lambda \|\mathbf{D}_i^j\|_0). \tag{53}$$

Note that since both $\mathbf{V}_i$ and $\mathbf{H}_i$ are orthonormal, the optimal solution is obtained by hard-thresholding the components of $\mathbf{D} = {\mathbf{V}_i^*}^T \mathbf{X}_i^j \mathbf{H}_i$ with $\sqrt{\lambda}$.



(a) (b) (c) (d)

Figure 25: Comparison of separable transforms of MDST and MDDT. MDST of vertical prediction (mode 0) (a), MDDT of vertical prediction (mode 0) (b), MDST of horizontal prediction (mode 1) (c), MDDT of horizontal prediction (mode 1) (d).

IV. *Optimal horizontal transforms for given coefficients:* Similarly, the optimal horizontal separable orthonormal transform can be calculated with updated coefficients and the vertical transform found in previous steps;

$$\mathbf{H}_i^* = \arg\min_{\mathbf{A}} \left\{ \sum_{\mathbf{X}_i^j \in S_i} \|\mathbf{X}_i^j - \mathbf{V}_i^* \alpha_i^{j*} \mathbf{A}^T\|_2^2 \right\} \tag{54}$$

$$s.t. \quad \mathbf{A}^T\mathbf{A} = \mathbf{I}.$$

Note this time vertical separable transform $\mathbf{V}_i$ is assumed to be fixed. Next, let $\mathbf{Y} = \sum_{\mathbf{X}_i^j \in S_i} \alpha_i^{j^T} \mathbf{V}_i{}^T \mathbf{X}_i^j$, and its SVD be $\mathbf{Y} = \mathbf{U}\Lambda^{1/2}\mathbf{W}^T$. The solution for the optimal orthonormal transform is $\mathbf{H}_i^* = \mathbf{W}\mathbf{U}^T$.

Return to Step I and repeat the process till the cost function converges to a steady state value. Sample transforms are shown in Figure 25 together with their MDDT counterparts. This optimization method differs from those used in [63] and [67]. In [63], the proposed transform design method reduces the sparsity-distortion cost of a set of data extracted from natural images via iterative clustering and transform optimization for the nonseparable case. In this chapter, the data is residual blocks extracted from a video coder, and the corresponding clusters are defined by the intra prediction mode. Hence, the data clusters are fixed, so relabeling after the transform optimization is not needed. Thus, the mode-dependent term is coined for the transforms in the current design.

The 2-D separable transform design provided in [67], which is based on the optimization given in [63], lacks the mode-dependent characteristic, and its iterative optimization has two shortcomings. The first problem is the update step for vertical transforms in Equation (9) of [67], whereby the vertical and horizontal separable components converges to the same transforms. For mode-dependent transforms, it is expected that the vertical and horizontal transforms will be different due to the directional characteristics of the residual data. Correction is provided in Step II of our iterative optimization procedure. The second problem stems from the iterative update of vertical and horizontal transforms without updating coefficients. When vertical or horizontal transforms are updated, the coefficients do not belong to new transform anymore. Therefore, in the iterative optimization described above, the transform update is always followed by a coefficient update.

5.4 Reordering Transforms

Entropy coders in current video codecs are optimized to work with the DCT. Although the optimization described in this paper initializes transforms with DCT, the resulting transform coefficients may compact energy in a different order than with the DCT. Therefore, the columns of the vertical and horizontal 2-D separable transform are reordered depending on the energy of the coefficient values of the residual data set of the corresponding mode.

Let $\mathbf{Q}$ be an $N \times N$ matrix whose entries are defined as follows:

$$\mathbf{Q}(m,n) = \sum_{j \in \mathcal{S}} \alpha^j(m,n)^2. \tag{55}$$

where α^j is the coefficient matrix of the j-th block in the training set of mode $\mathcal{S}$. Then sum of the energies along the rows and columns can be defined respectively as,

$$q_r(m) = \sum_n \mathbf{Q}(m,n) \ \ \forall m, \quad q_c(n) = \sum_m \mathbf{Q}(m,n) \ \ \forall n. \tag{56}$$

To rank these energies, the x and y variables that will satisfy

$$q_r(x_1) \geq q_r(x_2) \geq \ldots \geq q_r(x_N), \quad q_c(y_1) \geq q_c(y_2) \geq \ldots \geq q_c(y_N) \tag{57}$$

can be found. Next the columns of the optimized vertical and horizontal separable transforms are reordered as

$$\mathbf{V}^o(m,n) = \mathbf{V}(m,x_n), \;\; \mathbf{H}^o(m,n) = \mathbf{H}(m,y_n) \quad \forall m,n \tag{58}$$

where $\mathbf{H}$ and $\mathbf{V}$ become $\mathbf{H}^o$ and $\mathbf{V}^o$ after reordering. The new order statistically ensures that the coefficients with higher energy appear closer to the top-left corner of the coefficient matrix similar to DCT. The transforms for each mode and block size are ordered in same fashion. Later, they are scaled up and rounded off to have integer values.

5.5 Results

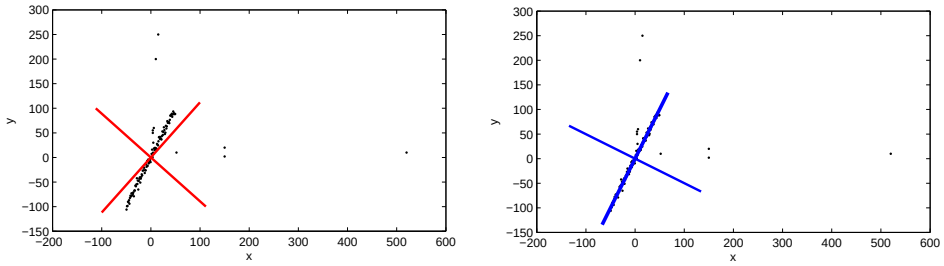
Two sets of experiments are provided in this section. First, the robustness of the $\mathcal{L}_0$-norm regularized solution is compared with KLT for a linear regression problem. The second set of experiments show the video coding performance of the proposed MDST method with respect to MDDT, which is already implemented in the JM11.0KTA2.6r1 (or KTA) codec. In addition, a set of KLT-based 2-D separable transforms is trained by using the same method as MDDT but with our data set. This enables us to analyze the effect of training data for the performance improvement that we achieved.

5.5.1 Model-based Experiment on Robust Regression

In this part a simple linear regression application of KLT and $\mathcal{L}_0$-norm regularized solution is given. A 2-D set of vector are generated by the following model

$$y = 2x + 5w \tag{59}$$

where w is a zero-mean and unit-variance Gaussian random variable. One-hundred Gaussian noise samples are generated and added to x values from -50 to 50. Both



(a) (b)

Figure 26: Crosses show axises of components found by KLT (a), and $\mathcal{L}_0$-norm regularized solution (b).

the KLT and the proposed $\mathcal{L}_0$-norm regularized solutions recover the correct principal direction. However, when random sparse outliers are included in the data set, the KLT fails to capture the correct direction, as shown in Figure 26(a). The $\mathcal{L}_0$-norm regularized solution, however, almost perfectly aligns with the direction of correlation set in (59), as shown in Figure 26(b). The only disadvantage of $\mathcal{L}_0$-norm regularized solution over KLT is its complexity, which is in general less of a concern for off-line training. Nevertheless, initializing the algorithm with the components of KLT and annealing λ improves the convergence speed. For this experiment, the cost function converged in 15 iterations with a fixed $\lambda = 50^2$ when the components were initialized with KLT.

5.5.2 Video Coding with MDST

The transforms generated by the proposed algorithm is used to replace the MDDT transforms currently implemented in the KTA software. As mentioned before, the transforms are trained by extracting intra prediction residuals for 4×4, 8×8, and 16×16 block sizes. For each block size, a set of 2-D separable transforms is trained with the proposed iterative optimization scheme described in Section 5.3. The training data contains High-Definition (HD) and CIF (352×288) sequences. The video frames used for training are not used for testing. The sequences are encoded as all intra pictures using four QP values 25, 29, 33, and 37. A different set of values are used

for HD sequences: 25, 28, 31, and 34. These QP values are identical to those used in [39]. The CABAC entropy coder is used, and the anchors used for comparison are generated using the DCT-enabled KTA encoder.

Table 2 shows experimental results for several sequences. To understand how much of the performance improvement comes from the data that is used to learn transforms, a set of controlled experiments are performed with KLT as the learning method, using the same set of training data. One could expect similar coding results between our controlled experiment and MDDT, provided that the training data of both are similar. For performance comparisons, BD metrics are used [8]. The negative value of BD-Rate means the tested method decreases the bitrate with that percent compared to the anchor method, which is H.264/AVC in these experiments. The positive BD-PSNR means the tested methods performs better than the anchor. On the second and third columns of Table 2, although BD-rate improvements of individual sequences differ, the averages are very close (our implementation is only 0.2% better). The fourth column shows performance improvement of the proposed method, MDST. Overall a BD-rate improvement of 1.29% is achieved over MDDT at *no extra cost*, and the improvement goes up to 2.97% in HD sequences.

Table 2: Coding performance, reference is JM-KTA 2.6r1

		MDDT			**KLT**			**MDST**			**KLT HD**			**MDST HD**		
Sequences	***number of frames***	**BD-Rate (%)**	**BD-PSNR (dB)**	**Avg BD-Rate**	**BD-Rate (%)**	**BD-PSNR (dB)**	**Avg BD-Rate**	**BD-Rate (%)**	**BD-PSNR (dB)**	**Avg BD-Rate**	**BD-Rate (%)**	**BD-PSNR (dB)**	**Avg BD-Rate**	**BD-Rate (%)**	**BD-PSNR (dB)**	**Avg BD-Rate**
352x288				**-5.07**			**-5.07**			**-5.73**			**-5.20**			**-5.52**
Foreman	100	-5.93	0.322		-5.84	0.318		-8.10	0.446		-6.16	0.335		-7.35	0.402	
Mobile	100	-4.27	0.444		-4.74	0.495		-4.77	0.500		-4.52	0.471		-4.77	0.499	
Coastguard	100	-5.39	0.335		-5.00	0.310		-5.09	0.315		-5.39	0.337		-4.86	0.301	
Container	100	-4.67	0.301		-4.71	0.305		-4.94	0.320		-4.74	0.307		-5.08	0.329	
832x480				**-5.07**			**-5.18**			**-6.26**			**-5.48**			**-6.01**
BasketballDrill	30	-5.87	0.288		-6.14	0.304		-6.91	0.344		-6.51	0.322		-6.90	0.342	
PartyScene	30	-3.85	0.294		-4.09	0.313		-5.31	0.408		-4.14	0.317		-4.88	0.376	
BQMall	30	-5.59	0.362		-5.42	0.353		-7.21	0.474		-5.96	0.389		-6.78	0.445	
RaceHorses	30	-4.69	0.306		-5.08	0.333		-5.61	0.370		-5.30	0.348		-5.48	0.361	
1920x1080				**-5.72**			**-6.09**			**-7.54**			**-7.22**			**-8.14**
Kimono1	10	-6.58	0.245		-6.65	0.249		-8.90	0.341		-9.68	0.372		-10.25	0.397	
ParkScene	10	-6.38	0.298		-6.96	0.327		-7.12	0.334		-6.99	0.329		-7.07	0.332	
Cactus	10	-6.39	0.257		-6.96	0.281		-7.70	0.312		-7.49	0.304		-8.05	0.326	
BasketballDrive	10	-4.47	0.114		-4.73	0.121		-7.44	0.192		-6.10	0.157		-8.38	0.217	
Tennis	10	-4.80	0.154		-5.16	0.167		-6.52	0.211		-5.86	0.191		-6.94	0.226	
Average				**-5.30**			**-5.50**			**-6.59**			**-6.07**			**-6.68**

For visual quality comparison, frames coded at the same rate using MDDT and

MDST are also provide in Figure 27. MDST result on the right have slightly better reconstruction of facial features and edges compared to MDDT.

Due to increased importance of efficiently coding HD sequences, one last set of experiments is done by changing training data set to all HD sequences. Both KLT of controlled experiment and MDST are learned from this new data. Columns five and six of Table 2 shows these results. Surprisingly, the KLT in this case has significant performance improvement not just on HD but for all the sequences. This can be attributed to the statistics of the residuals extracted from HD sequences. Compared to previous training data, it is likely that these residuals have fewer outliers, hence the components of KLT align better with the data. On the other hand, the training method used for MDST outperforms KLT-based learning in all these settings.

5.6 Conclusions

This paper presents the Mode-Dependent Sparse Transform (MDST), a new 2-D separable transform design for video coding. The implicit relation between sparsity-enforced optimization of transforms and robust learning is revealed. When the training data has outliers, the proposed training method is more robust than the conventional KLT-based training. Utilizing this approach, a new set of 2-D separable transforms are trained using residual data from each intra prediction mode in the KTA codec. Compared to DCT and MDDT-based video coding, bitrate reductions of up to 10.2% and 3.9% are achieved, respectively.

(a) (b)

Figure 27: Reconstructed foreman image (a) with MDDT 32.93dB and 0.196bpp, and (b) with MDST at 33.04dB and 0.194bpp

CHAPTER VI

RISK-MINIMIZING TRANSFORMS FOR SIGNAL ESTIMATION

6.1 Introduction

As a test bed for a diverse set of signal reconstruction problems, the signal denoising helps researchers to develop better signal representations. The core of the denoising problem is to extract a signal, which is embedded in a noise. Solutions to remove the noise from the signal can be as simple as applying median or mean filtering. In more sophisticated methods, researchers devised shrinkage methods and formulated their performance bounds for noise removal [26, 27], utilized various transform-domain representations [30, 35, 19], or analyzed the structural correlations within the signal and its transform-domain representations [35, 59, 72].

In this chapters, denoising with transform-domain representations is discussed with applications to noise removal for images. With the assumption that the energy of the noise is less than the embedded signal, the transform-domain representation separates the signal and the noise from the observed data. Later, the part of the presentation that corresponds to noise is removed and embedded signal is reconstructed by returning to the original signal domain. This denoising approach is also known as transformation-shrinkage-inverse transformation (TSI) structure. From the very beginning of the classical signal estimation theory, TSI structure exists within the Wiener filtering, if the principal components of the signal are used for the transform representation[11]. With a different perspective, the whole TSI structure works as an estimator, which reconstructs the original signal from its noisy and corrupted observation. Provided that the signal is well approximated by a transform, one can

get close to the performance of an ideal estimator by simply hard-thresholding the coefficients of that transform, which makes TSI structure more appealing [26]. Basically, the hard-thresholding operation keeps only the coefficients that are above some energy level (or a threshold).

The orthonormal transforms such as wavelets, cosine packets, the Karhunen-Loeve transform, and the DCT have long dominated the denoising literature because of their signal decorrelating properties. Exploiting the statistical properties of the natural images, sparse representations have recently been utilized to offer a new perspective to the problem [36, 30]. In [30], Elad et al. developed a sparse and redundant representation by training an overcomplete dictionary (K-SVD) for each image for denoising. This method (at the time of its publication) yielded the state-of-the-art performance in denoising. The strength of K-SVD as compared to standard orthonormal transforms is due to its ability to generate wide selection of structurally different atoms, which enables sparser representation at signal singularities. Our technique that is described in previous chapter, on the other hand, supplies a library of orthonormal basis functions that are trained in a sparsity-distortion optimal fashion to adapt to signal singularities [63]. It should be noted that the SOT representation is not redundant, yet its library provides a variety of structurally different orthonormal subspaces. With sparsity-distortion optimal transform adaptation, SOT yields an efficient signal representation. In this chapter, the estimation efficiency of the new optimization methods is examined. Later, we will show that the iterative training method of SOT converges to a set of orthonormal transforms that minimizes the estimation error over a class of signal, such that the shrinkage and transform designs procedures are coupled for better denoising.

After optimizing the transforms, a new image denoising method is proposed based in the translation-invariant denoising idea (or cycle-spinning)[18]. The cycle-spinning can be implemented by using redundant transforms (such as overcomplete DCT or

wavelet transforms). Another way to implement translation-invariant denoising is to denoise local neighborhood of each pixel in the image then take the average of the overlapping denoised estimates. For block transforms, this naturally helps to remove the blocking artefact that might show at the block boundaries. Guleryuz [35] replaced this standard averaging with a weighted averaging, and achieved very successful image denoising results. In this chapter, we will also address the weighted averaging notion in denoising and present the optimal fusion method of local estimates to generate the global and final signal reconstruction.

6.2 Estimating Signals in the Presence of Noise

In the previous chapters, we have focused either designing efficient signal representation or improving the efficiency of the existing ones. The emergence of wavelets, which is followed by the wavelet shrinkage, is a good example of the phenomenon of having an efficient representation leads to an efficient estimation [25, 24]. In his paper [11], Candes provides a clear and thorough explanation for this connection between the estimation and the representation problems. In this line of thought, the improved efficiency in representation by SOT is expected to have a reflection in the estimation problems. As the next logical step for the analysis of the proposed method, in this section we intend to analyze the implicit relation of signal representation and estimation provided by Sparse Orthonormal Transforms formula. First, the theory on signal estimation in the presence of noise is provided. Then we explain why the proposed learning method is relevant to estimation problem. As an implementation test bed, the image denoising problem is selected.

With the proposed learning method for orthonormal transforms, we claim that the blocks of image/video data in the training set are well approximated by the new set of orthonormal transform. Thus, one can expect that a simple thresholding operation,

which maps the coefficients of the orthonormal transforms to zero or leaves them unchanged, would give estimation results almost as good as the optimal estimators[46].

Starting from an observation model for the noisy observation:

$$\mathbf{y} = \mathbf{x} + \mathbf{w} \tag{60}$$

where $\mathbf{y}$ is the noisy observation, $\mathbf{x}$ is the underlying signal that will be estimated, and $\mathbf{w}$ is i.i.d. Gaussian noise $\mathcal{N}(0, \sigma^2)$. The notation for the transform-domain observation model for a transform Φ becomes

$$\begin{aligned} \beta = \Phi^T \mathbf{y} &= \Phi^T \mathbf{x} + \Phi^T \mathbf{w} \\ &:= \alpha + \eta. \end{aligned} \tag{61}$$

One can formulate denoising as a maximum-a-posteriori (MAP) estimation problem, which tries to increase the conditional probability of the original signal $\mathbf{x}$ given the noisy observation $\mathbf{y}$, i.e., $\mathbf{p}(\mathbf{x}|\mathbf{y})$. In line with the probabilistic framework of Section 2.2, the MAP estimator for denoising can be written

$$\hat{\alpha} = \arg\min_{d} \|\mathbf{y} - \Phi d\|_2^2 + \lambda \|d\|_p \tag{62}$$

where Φ can be a redundant or a non-redundant transform, and $\hat{\alpha}$ is the coefficients of the denoised estimate of $\mathbf{x}$, which is found by $\hat{\mathbf{x}} = \Phi^T \hat{\alpha}$. The methods to solve this minimization depend on the regularization norm, p, and the characteristics of the transform, Φ. In their paper [30], Elad and Aharon uses orthogonal matching pursuit[55] to solve this problem for an overcomplete dictionary (transform), where $p = 0$. Later, the overcomplete dictionary is updated based on denoised observation, and this procedure iterates till the dictionary and denoised signal reach to a stable state.

The solution to Equation (62) for $p = 0$ and an orthonormal Φ is hard-thresholding the coefficients. In the hard-thresholding method, i-th coefficient of the observation vector, $\beta(i)$, is kept or zeroed with respect to a threshold level $\tau = \sqrt{\lambda}$. Note that this

is exactly the same process used in the coefficient update step of the SOT optimization (refer to Algorithm 1).

The hard-thresholding operation applies the following weights to coefficients of observed (noisy) signal,

$$s(i) = \begin{cases} 1, & |\beta(i)| \geq \tau \\ 0, & |\beta(i)| < \tau. \end{cases} \tag{63}$$

One can form a diagonal matrix $\mathbf{S}$ with these weights as follows:

$$\mathbf{S} = \texttt{diag}([s(1), \cdots, s(n)]. \tag{64}$$

The diagonal matrix $\mathbf{S}$, which is also called as a selector matrix for hard-thresholding, is used to estimate the coefficients of the original signal from the coefficients of the noisy observation $\hat{\alpha} = \mathbf{S}\beta$. In the signal domain, this process corresponds to $\hat{\mathbf{x}} = \Phi\hat{\alpha} = \Phi\mathbf{S}\Phi^T y$, where the operator $D = \Phi\mathbf{S}\Phi^T$ is called an estimator (note the TSI structure).

In general, an estimator $\mathbf{D}$ maps the observations $\mathbf{y}$ to the denoised estimates $\hat{\mathbf{x}}$

$$\hat{\mathbf{x}} = \mathbf{D}\mathbf{y}. \tag{65}$$

In this chapter, we will focus on a special type of estimators called diagonal estimator. These estimators operate in transform domain and treats the coefficients of the observed signal independently to find the denoised estimate as described above. In general, the diagonal entries of $\mathbf{S}$ can take any real value, but for hard-thresholding, they can either be zero or one. This diagonal estimator is a denoising operator with transformation-shrinkage-inverse transformation structure. Depending on the transform and the selection (shrinkage/thresholding) method used, the estimation efficiency of denoising estimators changes. The estimation performance of these estimators is measured by their risks (or mean square error, MSE). The notation we have used for the risk of estimating the signal $\mathbf{x}$ with the estimator $\mathbf{D}$ is

$$R(\mathbf{D}, \mathbf{x}) = \mathbb{E}(\|\mathbf{x} - \mathbf{D}\mathbf{y}\|_2^2). \tag{66}$$

where the expectation is calculated over several noise realizations of the observation vector y. The goal of estimator design is to come close to the minimax risk, which is defined as the maximum risk of the best estimator, over a class of signal $\mathcal{S}$. Donoho and Johnstone [27] have showed that the diagonal estimators are nearly minimax, if the orthogonal transforms provide fast decaying coefficients. This means that if the signal is well approximated in a given orthogonal domain, then the diagonal estimators are nearly optimal.

With reflection of these observations, we will try to find a set of orthonormal transforms that will reduce the risk of estimation of a class of signal such as image blocks, $\mathbf{x}$'s. The idea is simple; if an estimator has lower risk, it is more likely to provide better estimation for the signal $\mathbf{x}$. However, since the original signal $\mathbf{x}$ is required to calculate the actual risk given in Equation (66), an approximation or some bound on the value of the risk is needed so that the transform-domain representation can be updated accordingly.

Fortunately, there are theoretical upper and lower bounds for the risk $R(\mathbf{D}, \mathbf{x})$ that will help us to improve the performance of estimators. Alternatively, one can formulate an unbiased estimate for the risk as well, which will be useful in our later discussions on denoising (details are provided in the next section).

The lower bound or the minimum thresholding risk of the estimator $\mathbf{D}$ is given as

$$R_o(\mathbf{D}, \mathbf{x}) = \sum_{i=1}^{N} \min(\alpha(i)^2, \sigma^2) \tag{67}$$

where σ^2 is the noise variance, and N is the signal dimension. This theoretical lower bound on the estimation risk is also called as the "oracle" risk, presuming that α values are provided by an oracle. The upper bound for the risk is formulated by Donoho & Johnstone [27] as follows: Let R be the thresholding risk with a threshold $T = \sigma\sqrt{(2log_eN)}$ then

$$R(\mathbf{D}, \mathbf{x}) \leq (2\log_e N + 1) \cdot (\sigma^2 + R_o(\mathbf{D}, \mathbf{x})). \tag{68}$$

Note that the thresholding risk is within certain proximity of the oracle risk and off only by a factor of $2log_eN$ (We refer interested readers to the resources provided by Mallat and Candes [46, 11], for a better treatment of the subject). In general, one can expect to reduce the estimation risk of a diagonal estimator (or thresholding risk) by lowering its oracle risk.

Utilizing the above observation, a learning algorithm is designed to improve the performance of estimators. Interestingly, a second look at Equation (16) reveals that, one can recast the transform optimization based on sparsity-distortion cost as an oracle risk minimization problem, which learns an orthonormal transform that will minimize the oracle risk of an ensemble of data

$$\min_{\Phi_k} \left\{ \sum_{\mathbf{x}^j \in \mathcal{S}_k} R_o(\mathbf{D}_k, \mathbf{x}^j) \right\} \quad s.t \ \Phi_k^T \Phi_k = \mathbf{I} \tag{69}$$

where $\mathbf{D}_k = \Phi_k \mathbf{S} \Phi_k^T$ is the diagonal estimator optimized for a class of data, $\mathcal{S}_k$. Also note that λ in Equation (16) is set to be equal to the noise variance, σ^2 for risk minimization. Basically, once the oracle risk of an estimator is minimized over a class of signal, one can expect improved estimation performance in the presence of noise.

6.3 Weighted Average Denoising Theory

To test the estimation performance with the new transforms, an image denoising algorithm is designed and implemented. Let $\mathbf{Y}$ be the noisy image of size $K \times K$,

$$\mathbf{Y} = \mathbf{X} + \mathbf{W} \tag{70}$$

where $\mathbf{X}$ is the original signal, and $\mathbf{W}$ corresponds to i.i.d Gaussian noise with zero mean and variance σ^2. The proposed denoising algorithm uses translation-invariant denoising idea (or cycle-spinning)[18], which is implemented by denoising overlapping blocks of pixels. To be more specific, let $\mathbf{x}_1$ and $\mathbf{x}_N$ be the first and the last blocks that include m-th pixel of image $\mathbf{X}$ in raster scan order as shown in Figure 28. For each block $\mathbf{x}_k$ for $k = \{1..N\}$ in the neighborhood of m-th pixel is denoised with an

estimator. Later, these N different estimate for the m-th pixel is fused to find the final pixel reconstruction.

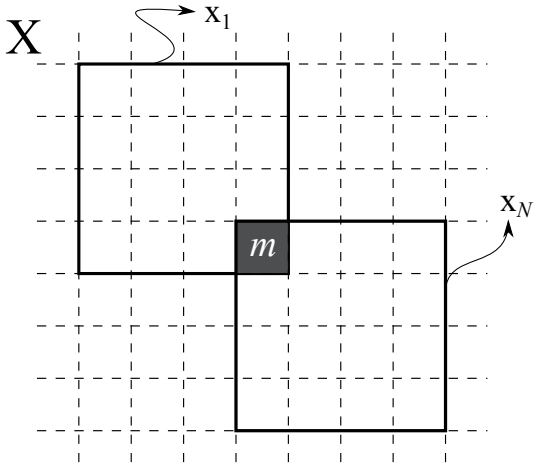


Figure 28: The first and last blocks ($\mathbf{x}_1$ and $\mathbf{x}_N$) in the raster scan that include the m-th pixel of the image $\mathbf{X}$.

Note that each transform defines an estimator, therefore there are K estimators for denoising. Among these estimators, the one that gives minimum risk for a given block of pixel is used for denoising

$$k = \arg\min_{\gamma} \left(R(\mathbf{D}_\gamma, \mathbf{x}) \right). \tag{71}$$

This means that the k-th estimator (and corresponding transform Φ_k) is the "best" denoiser for the block $\mathbf{x}$. Since actual risk is not available, to implement this decision method for the selection of estimators, we will use Stein's unbiased risk estimator (or SURE). Due to its simplicity, we prefer to use SURE for soft-thresholding risk given in [27],

$$SURE(\mathbf{D}, \mathbf{y}) = N - 2 \times \#\{i : |\beta i| < T\} + \sum_i \min(\beta(i)^2, T^2) \tag{72}$$

where β is the coefficient vector of the transform Φ for the noisy block $\mathbf{y}$, and T is the threshold. Stein's risk estimation is unbiased because $\mathbb{E}(SURE(\mathbf{D}, \mathbf{y})) = R(\mathbf{D}, \mathbf{x})$. In the implementation, "best" estimator for block $\mathbf{y}$ is found by

$$k = \arg\min_{\gamma} \left(SURE(\mathbf{D}_\gamma, \mathbf{y}) \right). \tag{73}$$

In cycle-spinning, the average value of the denoised estimates are used as the final reconstruction. Here we are taking a different route to fuse these estimates by formulating the optimal weight for each estimate. Previously, Guleryuz has shown that a weighting strategy to fuse denoised estimates of a DCT-based estimator can accomplish competitive results with highly complex wavelet-based denoising methods[35]. His work reveals that among the several denoised estimates for a pixel, there are some that are better than the others. Following this idea, in the next couple of sections we will formulate the optimal weighting function for the best final signal reconstruction in mean-square error sense.

6.4 Local to Global: Optimal Fusion of Denoised Estimates

Let's formulate the optimal weights for denoising the m-th pixel of the noisy image $\mathbf{Y}$. In Figure 28, we have showed the first and the last blocks ($\mathbf{x}_1$ and $\mathbf{x}_N$) that include m-th pixel of the original image $\mathbf{X}$. A block $\mathbf{x}_l$ that includes m-th pixel can be estimated from noisy observation by

$$\hat{\mathbf{x}}_l = \mathbf{D}\mathbf{y}_l \tag{74}$$

where $\mathbf{y}_l$ is the block in the noisy image Y, whose location matches with the location of $\mathbf{x}_l$, and $\mathbf{D}$ is an estimator. The denoised image $\hat{\mathbf{X}}$ can be reconstructed by weighting the overlapping estimates at the m'th pixel

$$\hat{\mathbf{X}}(m) = \sum_{l=1}^{N} \omega_l \hat{\mathbf{x}}_l(\underline{m}). \tag{75}$$

Here ω_l is the weight for the l-th estimation of the m-th pixel. To simplify the notation, we have used $\underline{m}$ to denote the corresponding coordinate of m-th pixel within that particular block. The optimal weights should minimize the expected squared error and needs to have unit sum to preserve the mean value of the image. These translate to the following Lagrangian cost

$$\mathbb{E}[(\mathbf{X}(m) - \hat{\mathbf{X}}(m))^{\mathbf{2}}] + \lambda_m(\sum_{l=1}^{N} \omega_l - 1) \tag{76}$$

which is needed to be minimized. Note that since the weights have unit sum, one can substitute Equation (75) into the Equation (76) and get

$$\mathbb{E}[\left(\sum_{l=1}^{N}\omega_l(\mathbf{X}(m)-\hat{\mathbf{x}}_l(\underline{m}))\right)^2]+\lambda_m(\sum_{l=1}^{N}\omega_l-1). \tag{77}$$

If we define the error of the l-th estimation as $\mathbf{e}_l=\mathbf{x}_l-\hat{\mathbf{x}}_l$, then we have

$$\mathbb{E}[\left(\sum_{l=1}^{N}\omega_l\mathbf{e}_l(\underline{m})\right)^2]+\lambda_m(\sum_{l=1}^{N}\omega_l-1) \tag{78}$$

Provided that the estimation error between two different estimators are uncorrelated at the m-th pixel, i.e., $\mathbb{E}[\mathbf{e}_i(\underline{m})\mathbf{e}_j(\underline{m})]=0$ for $1\leq i,j\leq n$ and $i\neq j$, one can minimize the Equation (78) by taking its derivative with respect to ω_l and setting it equal to zero.

$$\omega_l=\lambda_m(2\times\mathbb{E}[\mathbf{e}_l(\underline{m})\mathbf{e}_l(\underline{m})])^{-1}. \tag{79}$$

Intuitively, the optimal weight for the l-th estimation of the m-th pixel is found to be inversely related with the expected value of the square of the estimation error at m-th pixel.

6.5 Weighted Average Denoising with Estimator Risks

To implement this weighting strategy, we propose an approximation to Equation (79) as follows:

$$\omega_l=C_m\gamma_l \tag{80}$$
$$\gamma_l=\left(\sum_{j=1}^{N}\mathbb{E}[\mathbf{e}_l(j)\mathbf{e}_l(j)]/N\right)^{-1}$$

where γ_l is an approximation to $(\mathbb{E}[\mathbf{e}_l(\underline{m})\mathbf{e}_l(\underline{m})])^{-1}$, which is calculated over the support of estimator used in l-th estimation, and the scaling constant C_m is found by

$$C_m=\frac{1}{\sum_i\omega_i} \tag{81}$$

which makes the sum of weights at the m-th pixel equal to one. Note, since the sum of the expectations is equal to the expectation of the sum, one has

$$\sum_{j=1}^{N} \mathbb{E}[\mathbf{e}_l(j)\mathbf{e}_l(j)] = \mathbb{E}[\mathbf{e}_l^T \mathbf{e}_l] = \mathbb{E}[\|\mathbf{x}_l - \hat{\mathbf{x}}_l\|_2^2] \tag{82}$$

where $\mathbb{E}[\|\mathbf{x}_l - \hat{\mathbf{x}}_l\|_2^2]$ is the MSE of estimating block $\mathbf{x}_l$ with the estimator $\mathbf{D}$. From Equation (66), we know

$$\mathbb{E}[\|\mathbf{x}_l - \hat{\mathbf{x}}_l\|_2^2] = R(\mathbf{D}, \mathbf{x}_l). \tag{83}$$

Thus, by substituting risk term into Equation (80), the approximation to the optimal weight is found to be

$$\omega_l = NC_m / R(\mathbf{D}, \mathbf{x}_l). \tag{84}$$

This formulation suggest us to weight the denoised estimates with the risk of the estimators that are used to generate them. Therefore, the estimators with higher risks will get lower weights. We will use Equation (72) to estimate $R(\mathbf{D}, \mathbf{x}_l)$.

6.5.1 Estimator Support Size Adaptation

The performance of the proposed estimation method can be further improved, provided that the transform size is adapted in line with the local characteristics of the data. Up to this point, a fixed support size is assumed for estimation. Consider that, two sets of transforms with different support sizes are available for noising the neighborhood of a pixel. The adaptation with SURE in Equation (72) is not going to work for finding the best estimator among the estimators with different support sizes.

Fortunately, Equation (84) guides us to the right adaptation method for estimators(or transforms) with varying support sizes. In the previous derivations, since the number of samples N and the scaling constant C_m are fixed, the approximation to the optimal weight given in Equation (84) states that higher weights will be given to the estimators with lower risk. With the support size adaptation, the value of N will

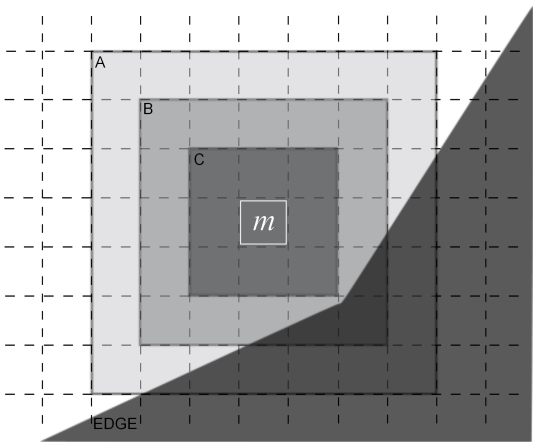


Figure 29: Three estimators with varying support sizes around an edge.Estimator A and C have largest and smallest support sizes, respectively.

vary for the estimators with different support sizes. While one can expect higher risk for estimator with larger support, the weight term is balanced by larger N value. To find the best estimator, the one with the highest weight is sought after as follows:

$$\{k^*, N^*\} = \arg\min_N \min_k \left[\frac{R(\mathbf{D}_k, \mathbf{x})}{N}\right]$$
$$for \quad \omega = NC_m/R(\mathbf{D}, \mathbf{x}) \tag{85}$$

Here, essentially the estimation risk is normalized by the support size of the estimator. In Figure 29, three estimators (A,B,C) centered at m-th pixel is shown. Note that the estimators A and B are crossing over the edge. Since it is difficult to represent an edge compared to smooth areas, the estimator C is likely the best estimator for denoising the m-th pixel. In the following section, we show an implementation of this estimator adaptation idea, which results with a powerful denoising method with globally learned dictionaries.

6.6 Implementation of Weighted Average Denoising

In the previous sections, we have given theory of the transform adaption, and formulated the optimal fusion of local denoised estimates to create the final image reconstruction. Here, a second iteration is added to the denoising method so that the

Table 3: Denoising performances of globally trained KSVD and SOT in terms of PSNR(dB).

	Lena		**Barbara**		**Peppers256**		**Boat**		**House**		**Cameraman**	
σ	**KSVD**	**SOT**	**KSVD**	**SOT**	**KSVD**	**SOT**	**KSVD**	**SOT**	**KSVD**	**SOT**	**KSVD**	**SOT**
2	43.21	**43.44**	42.42	**43.49**	42.63	**43.31**	41.74	**43.03**	44.27	**44.31**	42.43	**43.84**
5	38.47	**38.66**	37.20	**38.10**	37.62	**38.10**	36.64	**37.22**	38.83	**39.28**	37.46	**38.22**
10	35.41	**35.71**	33.06	**34.33**	34.28	**34.73**	33.53	**33.84**	35.65	**35.95**	33.49	**34.06**
15	33.60	**33.89**	30.60	**32.04**	32.34	**32.72**	31.62	**31.96**	34.03	**34.19**	31.32	**31.74**
20	32.25	**32.50**	28.86	**30.37**	30.92	**31.25**	30.22	**30.60**	32.81	**32.83**	29.84	**30.20**
25	31.19	**31.35**	27.57	**29.07**	29.78	**30.07**	29.15	**29.52**	31.78	**31.67**	28.73	**29.06**
30	30.31	**30.37**	26.56	**28.02**	28.86	**29.07**	28.29	**28.62**	30.89	**30.64**	27.87	**28.15**

calculations of the risks and the corresponding weights can be done more accurately. In the first iteration, an estimate to the original signal is generated as described in previous sections. This estimate is assumed to be close to the original signal, hence ideal estimation risk is calculated via the previous estimate (or one can use the oracle risk as well) for transform adaptation and weighted averaging. The outline of the iterative denoising methods is given as:

(1) : In the first iteration, for denoising the m-th pixel of the noisy image $\mathbf{Y} = \mathbf{X} + \mathbf{W}$

I For all blocks, $\mathbf{y}_l$, of size 8×8 that include the m-th pixel of $\mathbf{Y}$: Find the "best" estimator by minimizing SURE.

$$k = \arg\min_{\gamma}(SURE(\mathbf{D}_\gamma, \mathbf{y}_l)) \tag{86}$$

II Denoise each block with the diagonal estimator $\mathbf{D}_k$

$$\hat{\mathbf{x}}_l = \mathbf{D}_k \mathbf{y}_l, \quad l = \{1, .., N\} \tag{87}$$

where $\mathbf{D}_k = \Phi_k \mathbf{S} \Phi_k^T$. Depending on the coefficient values, β_k, of the transform Φ_k and the block $\mathbf{y}_l$, the diagonal entries of the selector matrix are found by

$$s(i) = \begin{cases} 1, & |\beta_k(i)| \geq \tau \\ 0, & |\beta_k(i)| < \tau. \end{cases} \tag{88}$$

where $\tau = \sqrt{(2log_eN)\sigma}$ for N and σ are the dimension of the signal and the standard deviation of the noise, respectively.

III Fuse denoised estimates to reconstruct the m-th pixel of the denoised image $\hat{\mathbf{X}}$

$$\hat{\mathbf{X}}(m) = \sum_l \frac{NC_m}{SURE(\mathbf{D}_k, \mathbf{y}_l)} \hat{\mathbf{x}}_l(\underline{m}) \tag{89}$$

where $\mathbf{D}_k$ is the "best" estimator for the block $\mathbf{y}_l$, and C_m is the scaling factor given in Equation (81).

(2) : In the second iteration, use $\hat{\mathbf{X}}$ for the selection of the diagonal estimators to denoise $\mathbf{Y}$.

I To denoise the m-th pixel, first find the "best" estimator via risk minimization

$$k = \arg\min_\gamma (R_{ideal}(\mathbf{D}_\gamma, \hat{\mathbf{x}}_l)) \tag{90}$$

where the blocks $\hat{\mathbf{x}}_l$ and $\mathbf{y}_l$ are colocated in the estimated and the noisy images, respectively. R_{ideal} represents ideal risk(refer to [46]), which is calculated by

$$R_{ideal}(\mathbf{D}_\gamma, \hat{\mathbf{x}}) = \sum_{\forall i} \frac{\hat{\alpha}_\gamma(i)^2}{\hat{\alpha}_\gamma(i)^2 + \sigma^2} \sigma^2. \tag{91}$$

Here $\hat{\alpha}_\gamma$ is the coefficient vector of the block $\hat{\mathbf{x}}_l$ with transform Φ_γ. It is also possible to use the oracle risk described in Equation (67) in here.

II With the "best" estimator find an estimate to the original signal

$$\tilde{\mathbf{x}}_l = \mathbf{D}_k \mathbf{y}_l, \quad l = \{1, .., N\} \tag{92}$$

where this time oracle attenuation is applied in the selector matrix, whose diagonal entries are

$$s(i) = \frac{\hat{\alpha}_k(i)^2}{\hat{\alpha}_k(i)^2 + \sigma^2}. \tag{93}$$

III Finally, fuse the estimates to reconstruct m-th pixel.

$$\tilde{\mathbf{X}}(m) = \sum_{l} \frac{NC_m}{R_{ideal}(\mathbf{D}_k, \hat{\mathbf{x}}_l)} \tilde{\mathbf{x}}_l(\underline{m}). \tag{94}$$

The experiments to test the denoising performance of the proposed method are conducted on a standard set of test images. First, five different realizations of Gaussian noise $\mathcal{N}(o, \sigma^2)$ are added to the original test images, then the proposed method is applied to reconstruct estimates to the original images. The support size of estimators are selected to be 8×8 and there are nine different estimators. Reported results are the average of the PSNR values of five different reconstruction. Table 3 shows comparison of the proposed denoising method and denoising with a globally trained overcomplete K-SVD dictionary in PSNR. SOT results are competitive even with image adaptive K-SVD.

Finally, using the estimator support size adaptation, the denoising algorithm presented above is updated. The only different between the fixed and adaptive support size is the replacement of the estimator selection methods in step(I) of both iterations given in the proposed iterative denoising method with the Equation (85). Estimators with three different support sizes are used for these experiments. The sizes are 4×4, 8×8, and 16×16 (same as Section3.4.4). Figure 30 shows PSNR difference versus the noise σ level. Denoising performance of the globally trained K-SVD is selected as the anchor method, therefore $y = 0$ line in Figure 30 represents K-SVD results in Table 3. In this figure, "K-SVD ADAPTIVE" denotes the performance of the image adaptive K-SVD denoising method, whereas the legend "SOT ADAPTIVE"is used to represent the proposed denoising algorithm with estimator support size adaptation. Finally, the results of "SOT" are from Table 3, where the estimator support size is fixed (8×8). To show that the PSNR improvements provided by the proposed denoising method reflect on the visual quality, Figure 31 to Figure 36 are provided below.

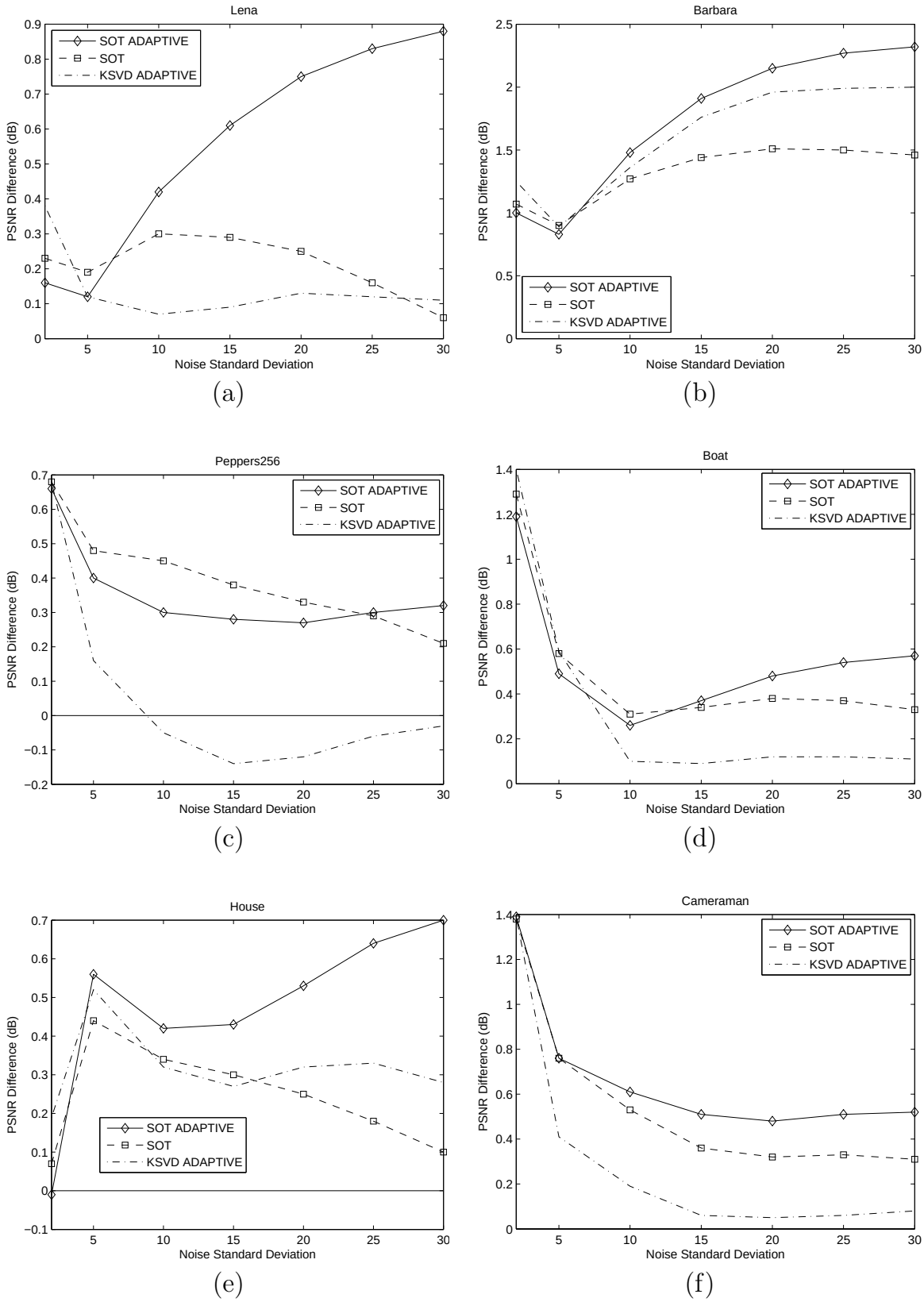


Figure 30: PSNR gains provided by denoising with sparse orthonormal transforms with fixed and adaptive support sizes (with legends "SOT" and "SOT ADAPTIVE") and image adaptive K-SVD ("KSVD ADAPTIVE") with respect to globally trained K-SVD denoising.

(a) (b)

(c) (d)

Figure 31: Original Lena image (a), Gaussian noise with $\sigma = 20$ is added (22.11db) (b), reconstruction of image adaptive K-SVD (34.41dB), reconstruction of proposed SOT with adaptive support size (33.00db).

Figure 32: Original Barbara image (a), Gaussian noise with $\sigma = 20$ is added (22.11db) (b), reconstruction of image adaptive K-SVD (30.82dB), reconstruction of proposed SOT with adaptive support size (31.00db).

(a) (b)

(c) (d)

Figure 33: Original Peppers256 image (a), Gaussian noise with $\sigma = 20$ is added (22.11db) (b), reconstruction of image adaptive K-SVD (30.80dB), reconstruction of proposed SOT with adaptive support size (31.19db).

(a) (b)

(c) (d)

Figure 34: Original Boat image (a), Gaussian noise with $\sigma = 20$ is added (22.11db) (b), reconstruction of image adaptive K-SVD (30.37dB), reconstruction of proposed SOT with adaptive support size (30.72db).

(a) (b)

(c) (d)

Figure 35: Original House image (a), Gaussian noise with $\sigma = 20$ is added (22.11db) (b), reconstruction of image adaptive K-SVD (33.12dB), reconstruction of proposed SOT with adaptive support size (33.36db).

(a) (b)

(c) (d)

Figure 36: Original Cameraman image (a), Gaussian noise with $\sigma = 20$ is added (22.11db) (b), reconstruction of image adaptive K-SVD (29.89dB), reconstruction of proposed SOT with adaptive support size (30.28db).

6.7 Conclusion

This chapter introduces how to use Sparse Orthonormal Transforms in a translation-invariant image denoising approach in the context of risk minimization. Basically, a neighborhood of each pixel is denoised by an estimator that uses transform - shrinkage -inverse transform structure. Then for the final signal reconstruction, an optimal weighting strategy is formulated to fuse the overlapping denoised estimates. With the estimator size adaptation, the proposed algorithm can produce the state-of-the art performance among the local denoising methods with lower computational requirements.

CHAPTER VII

CONCLUSION AND FUTURE WORK

This thesis presents a comprehensive treatment of dictionary learning problem for signal representation and estimation problems. For signal representation, a new method to design rate-distortion optimized orthonormal transforms is proposed. The fundamental idea behind using transform coding for signal representation is to exploit the regularity within data samples such that the redundancy of the representation is minimized subject to a fidelity cost. However, due to the non-stationarity of image, speech and audio signals the local statistics (hence the regularity) vary significantly across the data, which urges transform adaptation for efficient representation. Together with the transform optimization, this thesis couples the adaptation method with the corresponding optimization process to improve the efficiency of signal representation of orthonormal transforms. The image compression problem is selected as a test bed for understanding the performance of the new representation. First, the proposed method is used to generate sparsity-distortion-optimized orthonormal block transforms, which utilizes regularity along directional image singularities. These transforms are obtained by the joint optimization of the classification of blocks and the corresponding transforms of the classes over a training set. The result is a set of optimized transforms that replace the traditional block, lapped, and wavelet transforms used in image compression. Although the geometry information is used only at the initialization of the transform optimizations, the resulting transforms still retain directional structure.

For testing, a block-based codec is designed, which makes use of the new transforms for image coding. Consistent increase in bitrates compared to Discrete Cosine

Transform (DCT) based image codec is observed with up to 1dB improvement. In another block-based codec, the transform sizes are adaptively changed with a quadtree segmentation. Up to 2dB improvement is observed in natural images and up to 6dB improvement is observed for for synthetic images.

It is common to use surface representation and compression for 3-D geometry, where the geometry is mapped to an RGB image by resampling the data on a regular grid. These surface images of 3-D structures have strong geometric features , which can be efficiently represented with the proposed dictionary learning approach by designing a set of orthonormal transforms for this surface data. Application of the new transform learning method to the compression of 1-D and 3-D signals stands as the future works.

The proposed method in this thesis is a generic optimization method, which can be applied to improve coding efficiency of a variety of compression algorithms. To test this observation, a new lapped-transform-based codec is implemented that uses the proposed learning algorithm. Basically, on the top of the standard lapped bi-/orthogonal transform, a new set of directional transforms are learned. A consistent coding efficiency is gained over the standard lapped transform with up to 0.8dB improvement. This implementation is also one of the first directional lapped-transform designed in the literature, and we achieve this without getting into complex modulation techniques.

In wavelet-domain, similar to wedge- and foot-prints[28][74] ideas, the coefficients of wavelets are mapped to a sparser domain. Rather than using fixed models, a set of orthonormal transforms are designed and applied on the top of wavelet decomposition. This way, the regions that wavelet decomposition works are kept unchanged, while around the directional edges the new orthonormal transforms provide a sparser representation. Again a consistent increase in rate-distortion performance is observed compared to the original wavelet decomposition.

A new nonlinear wavelet decomposition algorithm is presented, which replaces the prediction step of the lifting algorithm with more complex 2-D interpolator that are designed to adapt the local context of the image. The local context is determined by extracting features from low-pass coefficients of the proposed decomposition algorithm. Similar to the interpolation with resolution synthesis method[5], a 2-D filter is learned for each context class. Subjective gains are observed around edges. For future studies, the same approach can be used as a new interpolation technique, which assumes the current image as the low-pass subband of a higher resolution image.

A novel separable filter design technique based on Chapter 3 is introduced for video coding. In the new design for each encoding mode a vertical and horizontal filter is learned by enforcing sparsity on the coefficients. The difference between the proposed transform design algorithm and Karhunen-Loeve transform (KLT) is explained based on robust statistics. This is done by examining the error norms of the KLT and the proposed method. We have revealed that due $\mathcal{L}_0$-norm regularization, the cost function (or the error norm, or M-estimator) of the proposed method reduces the influence of the outliers in the data. Robustness claims are supported by simple experiments provided in this chapter. When incorporated into a video codec, the new 2-D separable transforms are observed to produce state-of-the-art results. As a future work, the robustness of the new transform can be tested in different domains, for example in face recognition or object detection problems.

The improved efficiency in signal representation provided us a new and efficient way of signal estimation in noisy environments. To fuse these local denoised estimates, we have formulated the optimal fusion method of local estimates to generate the final denoised signal. We have shown that the sparsity-distortion cost is not only provides efficient signal representation and estimation but also the optimal weights for data fusion are also found to be inversely related with the sparsity-distortion cost.

Using the oracle risk, a risk minimization framework is described in Chapter 6.

The oracle risk of a diagonal estimator is used to find the upper and lower limits of the actual estimation risk. Once the oracle risk of an estimator is minimized over a class of signal, it is expected to improve the estimation performance. This is achieved by reformulation the original transform optimization given in Chapter 3 into a risk minimization problem, where we seek for transforms (any corresponding estimators) that will reduce the oracle risk.

With this framework, a set of transforms (or estimators) are learned and adaptively applied over a noisy data. The adaptation is done based on the risk of the estimators (the estimator that gives minimum risk value for that particular block of signal is selected). Together with this new adaptation, first the optimal fusion of local estimates is formulated and then a risk-based approximation is proposed to implement the new data fusion technique. Since we are using block transforms, the denoising operation (or estimation process) is performed per block. From local estimates, a global signal reconstruction is needed. Generally, averaging of the estimates are done to reconstruct the final denoised signal from the denoised blocks. Here, we have presented the theory and the implementation of optimal weighted averaging to improve the overall signal estimation efficiency. Also a formulation on how to fuse the estimators with different support sizes is given. The image denoising algorithm that is based on the new estimators and the adaptive support size selection shows significant estimation gains compared to dictionary-based denoising methods.

Motivated by the performance of signal estimation of the new denoising method, as a future work, several other inverse problems such as deblurring, inpainting and super-resolution can be solved. For this linear or nonlinear operators applied on the top of the original signal has to be incorporated into dictionary learning model. One example can be given for image deblurring, where a mapping is learned between the coefficients of the blurred image and the coefficients of the original signal.

REFERENCES

[1] Aach, T. and Kunz, D., "A lapped directional transform for spectral image analysis and its application to restoration and enhancement," *Signal Processing*, vol. 80, no. 11, pp. 2347–2364, 2000.

[2] Aharon, M., Elad, M., and Bruckstein, A., "K-svd: An algorithm for designing of overcomplete dictionaries for sparse representation," *IEEE Trans. on Signal Processing*, vol. 54, pp. 4311–4322, Nov 2006.

[3] Ahmed, N., Natarajan, T., and Rao, K., "Discrete cosine transfom," *IEEE Trans. Computers*, vol. C-23, no. 1, pp. 90–93, 1974.

[4] Antonini, M., Barlaud, M., Mathieu, P., and Daubechies, I., "Image coding using wavelet transform," *IEEE Trans. Image Proc*, vol. 1, pp. 205–220, April 1992.

[5] Atkins, C. B., Bouman, C. A., and Allebach, J. P., "Optimal image scaling using pixel classification," *in Proc. of IEEE International Conference on Image Processing*.

[6] Barlow, H., "Sensory mechanisms, the reduction of redundancy, and intelligence, the mechanisation of thought processes," *H.M.S.O., London,*, pp. 535–539, 1959.

[7] Bell, A. J. and Sejnowski, T., "The 'independent components' of natural scenes are edge filters," *Vision Research*, vol. 37, no. 23, pp. 3327–3338, 1997.

[8] Bjontegaard, G., "Calculation of average PSNR differences between RD curves," *ITU-T SG16/Q6, 13th VCEG meeting, Austin, Texas, USA*, pp. Doc. VCEG–M33, April 2001.

[9] Black, M. J. and Anandan, P., "The robust estimation of multiple motions: Parametric and piecewise-smooth flow fields," *Computer Vision and Image Understanding*, vol. 63, pp. 75–104, March 1996.

[10] Black, M. J. and Jepson, A. D., "Eigentracking: Robust matching and tracking of objects using view-based representation," *International Journal of Computer Vision*, vol. 26, no. 1, pp. 63–84, 1998.

[11] Candes, E., "Modern statistical estimation via oracle inequalities," *Acta Numerica*, pp. 1–69, 2006.

[12] Candes, E. and Donoho, D., *Curvelets: A surprisingly effective nonadaptive representation of objects with edges*. Vanderbilt University Press, 1999.

[13] Candes, E. and Romberg, J., "Quantitative robust uncertainty principles and optimally sparse decompositions," *Foundations of Computational Mathematics*, vol. 6, no. 2, pp. 227–254, 2005.

[14] Chang, C.-L. and Girod, B., "Direction-adaptive discrete wavelet transform for image compression," *IEEE Trans. Image Proc*, vol. 16, pp. 1289–1302, May 2007.

[15] Claypoole, R. L., Davis, G., Sweldens, W., and Baraniuk, R. G., "Nonlinear wavelet transforms for image coding via lifting," *IEEE Trans. Image Proc.*

[16] Cohen, A., Daubechies, I., and Feauveau, J., "Bi-orthogonal bases of compactly supported wavelets," *Comm. Pure Appl. Math.*, vol. 45, pp. 485–560, 1992.

[17] Cohen, A., Daubechies, I., Guleryuz, O., and Orchard, M., "On the importance of combining wavelet-based nonlinear approximation with coding strategies," *IEEE Trans. on Information Theory*, vol. 48, pp. 1895–1921, July 2002.

[18] Coifman, R. R. and Donoho, D. L., "Translation-invariant de-noising," in *Wavelets in Statistics (A. Antoniadis and G. Oppenheim, eds)*, pp. 125–150, Springer-Verlag, 1995.

[19] Dabov, K., Foi, A., Katkovnik, V., and Egiazarian, K., "Image denoising by sparse 3d transform-domain collaborative filtering," *IEEE Trans. Image Proc*, vol. 16, pp. 2080–2095, Aug. 2007.

[20] Daubechies, I. and Sweldens, W., "Factoring wavelet transforms into lifting steps," *J. Fourier Anal. Appl.*, vol. 4, no. 3, pp. 245–267, 1998.

[21] Ding, W., Wu, F., Wu, X., Li, S., and Li, H., "Adaptive directional lifting-based wavelet transform for image coding," *IEEE Trans. Image Proc.*, vol. 16, p. 416427, February 2007.

[22] Do, M. N. and Vetterli, M., "The contourlet transform: an efficient directional multiresolution image representation," *IEEE Trans. Image Proc.*, vol. 14, pp. 2091–2106, Dec 2005.

[23] Donoho, D. L., *Smooth wavelet decompositions with blocky coefficient kernels.* Academic Press: Recent Advances in Wavelet Analysis, 1993.

[24] Donoho, D. L., "De-noising by soft-thresholding," *IEEE Trans. Info. Theory*, vol. 41, no. 3, pp. 613–627, 1995.

[25] Donoho, D. L. and Johnstone, I. M., "Ideal denoising in an orthonormal basis chosen from a library of bases," *CR Acad. Sci., Ser. I*, vol. 319, pp. 1317–1322, 1994.

[26] Donoho, D. L. and Johnstone, I. M., "Ideal spatial adaptation via wavelet shrinkage," *Biometrika*, vol. 81, pp. 425–455, 1994.

[27] Donoho, D. L. and Johnstone, I. M., "Adapting to unknown smoothness via wavelet shrinkage," *J. Amer. Statist. Assoc.*, vol. 90, pp. 1200–1224, 1995.

[28] Dragotti, P. and Vetterli, M., "Wavelet footprints: Theory, algorithms and applications," *IEEE Trans. Signal Proc.*, vol. 51, pp. 1306–1323, May 2003.

[29] Dremeau, A. and Herze, C., "An em-algorithm approach for the design of orthonormal bases adapted to sparse representations," *In Proc. of 35th IEEE Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP '10), Dallas, TX*.

[30] Elad, M. and Aharon, M., "Image denoising via sparse and redundant representations over learned dictionaries," *IEEE Trans. on Image Processing*, vol. 15, pp. 3736–3745, Dec 2006.

[31] Engan, K., Aase, S., and Hakon-Husoy, J., "Method of optimal directions for frame design," *IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 5, pp. 2443–2446, 1999.

[32] Field, D., "What is the goal of sensory coding?," *Neural Comput.*, vol. 6, pp. 559–601, 1994.

[33] Gerek, O. N. and Cetin, A. E., "Adaptive polyphase subband decomposition structures for image compression," *IEEE Trans. Image Proc*, vol. 9, pp. 1649–1660, October 2000.

[34] Gerek, O. N. and Cetin, A. E., "A 2-d orientation-adaptive prediction filter in lifting structures for image coding," *IEEE Trans. Image Proc*, vol. 15, pp. 106–111, Jan 2006.

[35] Guleryuz, O. G., "Weighted averaging for denoising with overcomplete dictionaries," *IEEE Trans. Image Proc*, vol. 16, pp. 3020–3034, Dec 2007.

[36] Hoyer, P. and Oja, E., "Image denoising by sparse code shrinkage," in *Intelligent Signal Processing*, IEEE Press, 2001.

[37] Hubel, D. and Wiesel, T., "Receptive fields, binocular interaction and functional architecture in the cats visual cortex," *J. Physiol. (London)*, vol. 160, pp. 106–154, 1962.

[38] Hubel, D. and Wiesel, T., "Receptive fields and functional architecture of monkey striate cortex," *J. Physiol. (London)*, vol. 195, pp. 215–243, 1968.

[39] JTC1/SC29/WG11, I., "Call for evidence on high-performance video coding (HVC)." ISO/IEC JTC1/SC29/WG11 N10553, April 2009.

[40] Kreutz-Delgado, K., Murray, J., Rao, B., Engan, K., Lee, T., and Sejnowski, T., "Dictionary learning algorithms for sparse representation," *Neural Computation*, vol. 15, no. 2, pp. 349–396, 2003.

[41] Kreutz-Delgado, K. and Rao, B., "Focuss-based dictionary learning algorithms," *In Wavelet Applications in Signal and Image Processing VIII*, vol. 4119, 2000.

[42] la Torre, F. D. and Black, M. J., "A framework for robust subspace learning," *International Journal of Computer Vision*, vol. 54, no. 1-3, pp. 117–142, 2003.

[43] Lesage, S., Gribonval, R., Bimbot, F., and Benaroya, L., "Learning unions of orthonormal bases with thresholded singular value decomposition," *In Proc. of 15th IEEE Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP '05)*, pp. 293–296, Mar 2005.

[44] Lewicki, M. and Olshausen, B., "A probabilistic framework for the adaptation and comparison of image codes," *Journal of the Optical Society of America A Optics, Image Science and Vision*, vol. 16, no. 7, pp. 1587–1601, 1999.

[45] Liu, Y. and Ngan, K. N., "Weighted adaptive lifting-based wavelet transform for image coding," *IEEE Trans. Image Proc.*, vol. 17, p. 500511, April 2008.

[46] Mallat, S., *A Wavelet Tour of Signal Processing: The Sparse Way*. Academic Press, 3rd ed., 2009.

[47] Mallat, S. and Falzon, F., "Analysis of low bit rate image transform coding," *IEEE Trans. Signal Proc.*, vol. 46, pp. 1027–1042, April 1998.

[48] Mallat, S. and Zhang, Z., "Matching pursuits with time-frequency dictionaries," *IEEE Trans. Signal Processing*, vol. 41, no. 12, pp. 3397–3415, 1993.

[49] Malvar, H. S., "Signal processing with lapped transforms," *Artech House*, 1992.

[50] Malvar, H. S., "Biorthogonal and nonuniform lapped transforms for transform coding with reduced blocking and ringing artifacts," *IEEE Trans. Signal Proc.*, pp. 1043–1053, Apr. 1998.

[51] Olshausen, B. A. and Field, D. J., "Natural image statistics and efficient coding," *Network: Computation in Neural Systems*, vol. 7, no. 2, pp. 333–339, 1996.

[52] Olshausen, B. A. and Field, D. J., "Sparse coding with an overcomplete basis set: A strategy employed by v1?," *Vision Research*, vol. 37, pp. 3311–3325, 1997.

[53] Olshausen, B. and Field, D., "Emergence of simple-cell receptive field properties by learning a sparse code for natural images," *Nature*, vol. 381, pp. 607–609, 1996.

[54] Ouaret, M., Dufaux, F., and Ebrahimi, T., "On comparing JPEG2000 and intraframe AVC," *SPIE Optics and Photonics, Applications of Digital Image Processing XXIX, San Diego, CA USA*, vol. 6312, 2006.

[55] Pati, Y. C., Rezaiifar, R., and Krishnaprasad, P. S., "Arthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition," *presented at the 27th Annu. Asilomar Conf. Signals, Systems, and Computers*, 1993.

[56] Pennebaker, W. B. and Mitchell, J. L., *JPEG: Still Image Compression Standard*. Van Nostrand Reinhold, 1993.

[57] Pennec, E. L. and Mallat, S., "Sparse geometric image representation with bandelets," *IEEE Trans. Image Proc*, vol. 14, pp. 423–438, April 2005.

[58] Peyre, G. and Mallat, S., "Discrete bandelets with geometric orthogonal filters," *In Proc. of 12th IEEE Int. Conf. on Image Processing*, pp. 65–68, Sept 2005.

[59] Portilla, J., Strela, V., Wainwright, M., and Simoncelli, E. P., "Image denoising using scale mixtures of gaussians in the wavelet domain," *IEEE Trans. Image Proc*, vol. 12, pp. 1338–1351, Nov. 2003.

[60] Said, A. and Pearlman, W. A., "A new fast and efficient image codec based on set partitioning in hierarchical trees," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 8, pp. 243–250, 1996.

[61] Sezer, O. G., Altunbasak, Y., and Guleryuz, O. G., "A sparsity-distortion optimized multiscale representation of geometry," *In Proc. of 17th IEEE Int. Conf. on Image Processing, Hong Kong*, Oct 2010.

[62] Sezer, O. G., Cohen, R., and Vetro, A., "Robust learning of 2-d separable transforms for next-generation video coding," *In Proc. of IEEE Data Compression Conference, Snowbird, UH*, pp. 63–72, March 2011.

[63] Sezer, O. G., Harmanci, O., and Guleryuz, O. G., "Sparse orthonormal transforms for image compression," *In Proc. of 15th IEEE Int. Conf. on Image Processing, San Diego, CA*, pp. 149–152, Oct 2008.

[64] Shapiro, J., "Embedded image coding using zerotrees of wavelet coefficients," *IEEE Trans. Signal Proc.*, vol. 41, pp. 3445–3462, July 1993.

[65] Shum, H., Ikeuchi, K., and Reddy, R., "Principal component analysis with missing data and its application to polyhedral object modeling," *IEEE Pattern Analysis and Machine Intelligence*, vol. 17, no. 9, pp. 855–867, 1995.

[66] Simone, F. D., Ouaret, M., Dufaux, F., Tescher, A., and Ebrahimi, T., "A comparative study of JPEG 2000, AVC/H.264, and HD photo," *SPIE Optics and Photonics, Applications of Digital Image Processing XXX, San Diego, CA USA*, 2007.

[67] Sole, J., Yin, P., Zheng, Y., and Gomila, C., "Joint sparsity-based optimization of a set of orthonormal 2D separable block transforms," *In Proc. of 16th IEEE Int. Conf. on Image Processing*, Nov 2009.

[68] Strang, G., *Linear Algebra and Its Applications*. Brooks Cole, 3rd ed., 1998.

[69] Sweldens, W., "The lifting scheme: A construction of second generation wavelets," *SIAM J. Math. Anal.*, vol. 29, no. 2, pp. 511–546, 1997.

[70] Taubman, D., "Adaptive, non-separable lifting transforms for image compression," *in Proc. of IEEE ICIP*, vol. 3, pp. 772–776, October 1999.

[71] Taubman, D. and Marcellin, M., *JPEG2000: Image compression fundamentals, standards and practice*. Boston, Kluwer Academic Publishers, 2rd ed., 2001.

[72] Tomasi, C. and Manduchi, R., "Bilateral filtering for gray and color images," in *IEEE ICCV*, pp. 839–846, 98.

[73] van Hateren, J. and van der Schaaf, A., "Independent component filters of natural images compared with simple cells in primary visual cortex," *Proc. R. Soc. London B*, vol. 265, pp. 359–366, 1998.

[74] Wakin, M., Romberg, J., Choi, H., and Baraniuk, R., "Wavelet-domain approximation and compression of piecewise smooth images," *IEEE Trans. Image Proc*, vol. 15, May 2006.

[75] Wiegand, T., Sullivan, G., Bjntegaard, G., and Luthra, A., "Overview of the H.264/AVC video coding standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, pp. 560–576, July 2003.

[76] Xiong, Z., Guleryuz, O., and Orchard, M. T., "A DCT-based embedded image coder," *IEEE Signal Processing Letters*, vol. 3, pp. 289–290, November 1996.

[77] Xu, L. and Yuille, A., "Robust principal component analysis by self-organizing rules based on statistical physics approach," *IEEE Transactions on Neural Networks*, vol. 6, no. 1, pp. 131–143, 1995.

[78] Ye, Y. and Karczewicz, M., "Improved H.264 intra coding based on bi-directional intra prediction, direction transform, and adaptive coefficient scanning," *In Proc. of 15th IEEE Int. Conf. on Image Processing*, Oct 2008.

VITA

Born in Istanbul, Turkey, Osman G. Sezer received his BS degree in Electrical Eng. from Bogazici University, Istanbul, Turkey with Honors. He holds MS degrees from both Sabanci University and Georgia Tech. He has completed his Ph.D. in Center for Signal and Image Processing at Georgia Tech. The focus of his research has been on various aspects of image/video processing from pattern recognition to data compression. He has mostly been involved with next-generation video/image compression algorithms by collaborating with researchers at MERL, Docomo USA Labs and Texas Instruments (TI) R&D DSP Center. He is the recipient of two student paper awards from SPIE Visual Comm. Image Processing and IEEE Signal Processing & Comm. Applications conferences and received research fellowship from TI Leadership University Program during his Ph.D studies.